DATA ACQUISITION BOARD FOR STROKE REHABILITATION HAND DEVICE

by

Jacob Carducci

A thesis essay submitted to The Johns Hopkins University

in conformity with the requirements for the degree of

Masters of Science in Engineering

Baltimore, Maryland

February, 2018

Abstract

Individuals with weakened upper limbs from stroke-related incidents need early rehabilitation with accurate tools and engaging applications to improve patient outcomes. In an effort to enhance a pre-existing version of a force-sensing rehabilitative tool, known tentatively as the Stroke Rehabilitation Hand Device, a new data acquisition board with support for up to 20 analog channels was designed and evaluated. By creating a PCB unit of the DAQ design for use in the Hand Device, adjusting and calibrating a patient's finger-generated forces for proper data processing became easier and more viable. From system validation testing which included exerting constant force over set intervals of time, it was found that the device can detect isometric microforces well below 1 N of force and exhibits below 5% repeatability error. The accurate and accessible nature of the device should result in quicker treatment and recovery for patients using the device.

Readers: Dr. Noah J. Cowan and Dr. John Krakauer

Acknowledgments

I would like to thank Dr. Kevin Olds for all of the guidance and support he was able to give me. I also would like to thank Dr. John Krakauer, Dr. Omar Ahmad, Promit Roy, Alex Forrence, and the rest of the BLAM Lab for their support throughout the project. Furthermore, I want to thank Dr. Russell Taylor and Alexis Cheng for their feedback and support during the CIS II course, and Dr. Noah Cowan for guidance and editorial support.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Motor-neurological injuries, especially stroke, have negatively impacted the quality of life for numerous individuals nationwide. Arguably one of the more debilitating injuries are lesions that affects the strength and control of the hands and upper limbs, which makes it difficult to perform everyday tasks including grabbing objects and interacting with computers.

Previous studies reported efforts to characterize the multidirectional forces exerted by fingertips of stroke patients. For instance, researchers at the Rehabilitation Institute of Chicago and Northwestern University observed the workspace of index fingers of stroke survivors to understand changes in kinematics from motor impairment (E. Cruz and Kamper, 2005). Index fingers were fitted into a fiberglass cast affixed to a 6-DOF load cell, which could detect isometric force generation in different directions and postures. 24 subjects with prior history of chronic hemiparesis fitted the index finger of their impaired hand, whereas 6 control subjects had the index finger of their dominant hand fitted. Prior to the trials, the stroke-affected patients were binned equally into 4 Chedoke-McMaster stages of increasing observed hand performance.

Each subject generated maximum force in one of six orthogonal directions for a duration of 5 seconds with 1 minute-long periods of rest. Additionally, researchers recorded muscle activity at specific points on the finger. Subsequent ANOVA and Tukey analysis suggests that individuals with greater impairment in active finger movement generate less overall kinetic force than less impaired individuals and control subjects. Stroke-affected subjects also exhibited directionally-dependent and posture-dependent impairment in force generation. Kinematically, these subjects cover a fraction of the theorectical workspace reachable with a index finger joint system, with individuals in Chedoke stages with higher prior performance covering more of the workspace on average than subjects in lower stages. Control subjects cover even more workspace on average than any of the Chedoke groups. This suggests that isometric forces can still be detected in post-stroke patients with affected upper-arm function, but it is less apparent than unaffected individuals.

In an effort to understand why deficits in upper-limb manipulation occurs, studies have been conducted to characterize changes in interaction. At the Cleveland Clinic Foundation and Pennsylvania State University, researchers developed an apparatus and task routines to measure and track finger force production in stroke-affected patients (S. Li, 2003). For their particular study, they enlisted 16 individuals with recent stroke incidents and 16 age and gender matched control subjects. Each subject was instructed to conduct a two-part experiment with the first half dealing with single-hand tasks and the latter half with dual-hand tasks. In single-hand tasks, subjects exerted maximal force onto piezoelectric sensors with individual fingers, with pairs of fingers,

and all four fingers at a time. Dual-hand tasks have subjects exert force with fingers on both hands, which may be symmetrical or asymmetrical across both hands depending on the task. For non-targeted or enslaving fingers, subjects were instructed to refrain from pressing or lifting said fingers. During one-hand tasks, all fingers and certain combinations of multiple fingers on the impaired hands of stroke patients exhibited lower maximum voluntary contraction (MVC) force than on their non-impaired hands, and exhibited lower MVC force compared to the matched hands of control subjects. Additionally, enslaving forces from non-target fingers were higher for impaired hands of individuals affected by stroke compared to their non-impaired hands and to control subjects. This suggests that not only do patients with impaired upper limb function have less finger strength, but they have a tougher experience with controlling and suppressing unneeded digits for a task.

There are several different methodologies to quantify the motor function of the human hand such as the Fugl-Meyer Assessment and the Action Reach Arm Test (A. Fugl-Meyer and Steglind, 1975) (Lyle, 1981). Such methods do not make clear distinctions between finger strength and individual finger control however. Researchers at Johns Hopkins' own Department of Neurology have recently completed a study into post-stroke recovery of the human hand, where they have explored how the strength and control of fingers improves over time for various degrees of stroke (J. Xu and Diedrichsen, 2016). Specifically, they attempted to develop novel metrics for strength and individuation through implementing a custom-built hand evaluation device. Fifty-four patients affected by ischemic stroke and hemiparesis and fourteen

healthy control participants across three medical centers volunteered for five separate visits over a period of fifty-four weeks. During each visit, a given participant has both hands fitted into custom keyboard devices which can track one-dimensional pressing forces from each digit. The participant is presented with a display of finger forces in the form of moving horizontal bars on vertical scales for each digit. Desired levels for the strength and individuation tasks were indicated by green bars with a fixed height per trial. Strength, or MVC, was evaluated twice for each digit by having patients depress and maintain a given finger at maximum strength for two seconds. Individuation was evaluated by having each digit reach and maintain one of four sub-MVF target levels for 0.5 seconds, repeating each level four times. Overall strength was averaged over all five digits and normalized into an index metric with respect to the non-paretic hand performance at the end of the study period. In theory, an Index value of 1 corresponded to full recovery. The index for individuation is the negative logarithm of the regression ratio between the active finger force and passive finger enslaving, which is then averaged across all active digits and normalized to end-study non-paretic performance. Perfectly immobile passive fingers across all possible active digits correspond to a zero-value regression slope and an Individuation Index value of 1. They tracked the maximum voluntary contraction force of each finger to measure strength, while they compared the force generated by the active finger relative to the passive fingers to quantify individuation/control. They found that both strength and control in these forms improve the most within the first twelve weeks of recovery. Therefore, it is key to introduce rehabilitation to the patient as early as possible to improve the chances and effectiveness of

recovery, which can lead to a better quality of life for the patient.

An important aspect of rehabilitation is the ability to keep stroke patients engaged and consistently changed over the course of their recovery. Jean-Claude Metzger et al. at ETH Zurich explored an adaptive therapy technique to understand the feasibility of maximizing participation and minimizing frustration through careful adjustment of difficulty (J. Metzger and Gassert, 2014). In a prior design study, they developed the ReHapticKnob, a 2-DOF passive haptic robot that can track the squeezing distance between the thumb and the index finger of a given hand (known as grip aperture) and the pronosupination rotation angle of the corresponding forearm (J. Metzger and Gassert, 2011). For the difficulty study, six subacute stroke patients with mild upper limb impairment and hemiparesis were selected. Four days per week, each patient received 45 minutes of neurocognitive therapy utilizing the ReHapticKnob. A given session had three 15-minute exercises selected from seven possible neurocognitive exercises. These exercises evaluated aspects of proprioception of grip aperture and pronosupination, haptic perception of stiffness from grasping and pinching, sensorimotor memory of grip aperture and pronosupination, and sensorimotor coordination through forearm rotation. Before the exercises started, patients also were instructed to complete three robotic assessments to determine the active range of motion, patient-specific difficulty levels, and the initial exercise difficulty. Difficulty for a given exercise was adjusted based on a patient's calculated performance from the last exercise, which in most cases was a percentage of successfully completed trials. Average performance across all sessions with adaptive difficulty was 64%, with

the highest average for a given session being 71%, the lowest being 57%, and the overall average in variability being $\pm 21\%$. The researchers demonstrated solid statistical evidence that the pronosupination range of motion, FMA-UE score, and compliance perception improved by the follow-up interview while proprioception initially worsened then improved over time. The findings support the notion that early and appropriate engagement contributes positively to hand performance in stroke patients.

Researchers at Gifu University in Japan have developed an 18-DOF robotic exoskeleton with multi-axis key sensors that can accommodate small motions from weak users for self-motion therapy (S. Ito and Abe, 2007). Each digit and the wrist are supported by motion assist mechanisms, where 3-axes force sensors can detect joint torques from the fingers and servo motors can provide appropriate assistive motion for the joints. As a result, the exoskeleton can support the flexion/extension and abduction/adduction of each hand joint, as well as the flexion/dorsiflexion of the wrist and pronation/supination of the forearm. The system also provides control via a separate sensing data glove on the user's healthy hand, allowing the impaired hand to mimic healthy motion symmetrically. Upon response testing, the designers found that all motions except thumb abduction/adduction had negligible error. When implementing self-control therapy on a single healthy subject by instructing them to follow reference positions, the researchers observed that actual and reference trajectories roughly matched. From the results, this powered exoskeleton solution seems promising for helping treat patients. However, the exoskeleton appears to be expensive to reproduce from its inherent complexity attributed in part

to the 22 servo motors and 3 separate CPUs needed to operate the device. The exoskeleton is also heavy due to the numerous metal components and motors needed to assemble the device, and the assembly might harm the user without clinical guidance, making home use impractical.

In terms of proper treatment, passive and active haptic devices usually possess positive aspects of rehabilitation that the other category is unable to achieve. For example, active robotics can control the level of force for patients, but are often bulky and have limited range; passive robotics are more compact and are associated with larger ranges of motion, but they possess no way to control the motions of the user's hand when needed. To maximize the positive characteristics and minimize the negative aspects within a single package for patients, Ludovic Dovat et al. have developed a hybrid device known as the HandCARE rehabilitation system (L. Dovat and Burdet, 2008). The device secures the fingers of patients to cables guided by pulleys, and is back-drivable to support opposing motions. To reduce force bias from non-collinear forces, differential sensing of cable tension is implemented through the use of a three-pulley system and force sensors positioned orthogonally to the cable. A clutch subsystem can modify the movement behavior of all five fingers from a single actuator. Three possible movement modes exist for a given finger: a free mode where the cable cogwheel is disengaged and the finger can drive the cable without restriction, an active mode where the cogwheel is engaged with a torque-generating motor/encoder subsystem, and a fixed mode where the cogwheel is fixed to induce cable blocking. To determine these characteristics, the authors recruited eight healthy participants and five right-handed chronic

stroke patients for a brief measurement study. The authors measured the orientations and lengths of fingertips for each individual, and concluded that post-stroke subjects had different orientations and trajectory amplitudes due to limited finger abduction. This reaffirms that individuals affected by stroke have different kinetic and kinematic patterns in finger movement. The device appears to be complex and bulky though, making it relatively unsuitable for home use and easy clinical use.

There are pre-existing solutions on the market that attempt to provide upper-limb rehabilitation, but they possess serious characteristic flaws that limit their effectiveness. The Amadeo system manufactured by Tyromotion relies on linear motion sensing and actuation to provide rehabilitation (Tyromotion, 2017). Users are fitted into a brace and have their fingers manipulate a linear slide controller in front of a computer workstation. Depending on the nature of the therapy, the therapist can opt for continuous external stimulation of the hand or active training at the user's performance limit. The product features various measurement programs for metrics such as force, tone, and range of motion. However, an apparent limitation of this solution would be that the system is not very portable due to its rolling workstation nature and would restrict it to a clinical or medical environment. Additionally, there is considerable time needed for the patient to secure their wrist and fingers to the device, and weaker patients may not be able to overcome the internal friction needed to move the finger beams.

Another commercial example is the Rapael from NEOFECT, which is a flexible, glove-like device that integrates bend sensors to detect the patient's

hand motion (NEOFECT, 2016). The product also features a 9-axis movement and position sensor for detecting wrist movements. Like the Amedeo, the Rapael processes and sends information from the sensors to a computer system that runs interactive applications for user engagement. The product is intended for training forearm, wrist, and finger movements in order to strengthen them. However, the bend sensors can detect only one direction and can still be too stiff for weaker patients to flex.

While the aforementioned devices do provide useful features for upper-limb rehabilitation, they do not offer 3-DOF sensing of isometric micro-forces for all digits of the human hand, which would assist patients with extreme hand weakness. The group I am collaborating with believes that a low-cost, portable, and easy-to-use device can be developed for effective rehabilitation both in a clinical and in a home environment. Their answer is the Stroke Rehabilitation Hand Device, which is realized currently as a preliminary prototype, shown in Fig. 1.1. The idea of the device is to provide light passive resistance to stroke-affected users with flexible retention cups, and force sensors underneath would collect relevant force information for use in engaging applications that adapt to the ability of the user. The original prototype utilized a NI USB-6001 DAQ board, shown in Fig. 1.2. This component of the Device provides an electronic bridge between the finger sensors and the computer, and can amplify and process force signals. The most significant limitation of the original implementation is that it can support up to eight analog channels. Since each finger sensor requires four independent channels, the NI board can read from a maximum of two fingers. The goal of the new prototype's data

acquisition subsystem is to increase the channel count from 8 to 20 channels to support all five digits of the typical human hand. To that end, a novel custom data acquisition board was developed with an accompanying API and modified finger boards.

Figure 1.1: Overall device view A user has their hand fitted into the first prototype of the Stroke Rehabilitation Hand Device which is connected to a laptop. The laptop is loaded with an interactive application.

Figure 1.2: Old DAQ board A NI USB-6001 DAQ board, with a breakout channel board for finger sensors, is attached underneath the old Hand Device prototype.

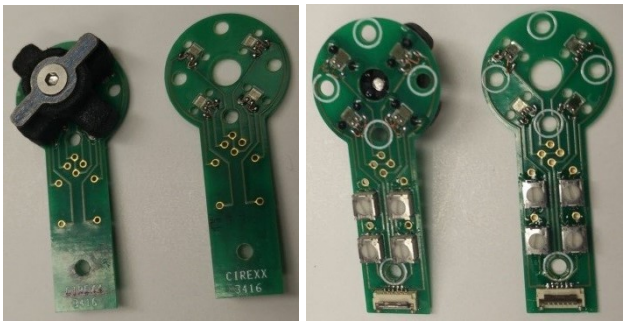
Chapter 2

Mechanical Design

In order to detect forces from a user's fingers, the finger interfaces are fitted with a custom PCB, as shown in Fig. 2.1.

The original force sensor design features four pairs of strain gages, four trimmer potentiometers, and a FPC connector. For this implementation, Micro-Measurements S5024 5K ohm linear patterns were used as strain gages, while 5K ohm Bourns Trimpots were used for the offset potentiometers and Hirose 6-position, 0.5 mm pitch FPC connectors are used for interfacing with the PCB. The board substrate is 1.0mm thick and is custom manufactured by Cirrex.

The four gage pairs are arranged in a square diamond configuration, as shown in Fig. 2.2. Two gages of a pair are located at specific points on the board on directly opposite sides to one another. The gages are electronically connected in a voltage divider setup, shown in Fig. 2.3. One gage is directly connected to an excitation voltage port and a signal port, while the other gage is connected to that same signal port and a ground port. As force deflects a gage pair location, one gage will experience an increase in resistivity while the other experiences a comparable decrease due to their opposing orientations.



(a) Top side **(b)** Bottom side

Figure 2.1: Original force sensor An unconnected force sensor features potentiometers for manually adjusting offset and an attachment frame for connecting silicone retention cups.

This can be expressed by Equations 2.1 and 2.2:

$$\pm\Delta V = 2i_{eff}\Delta R \tag{2.1}$$

$$2\Delta R = k_\epsilon F \tag{2.2}$$

where ΔR is the resistance change per gauge from a zero-strain resistance R, ΔV is the voltage difference change from a zero-strain voltage V, F is the equivalent point normal force experienced at the gage pair, and ϵ is the strain/force ratio of the PCB stemming from its physical properties.

This, ideally, results in twice the voltage difference at the signal port for a given current compared to a single gage, doubling the sensitivity of the circuit.

Furthermore, any two voltage divider circuits can be combined to create a Whetstone bridge by reading the voltage difference between two signal ports, which are labeled as 1, 2, 5, and 6 on Fig. 2.2. (Ports 3 and 4 correspond to input excitation and ground). From different combinations of signal ports,

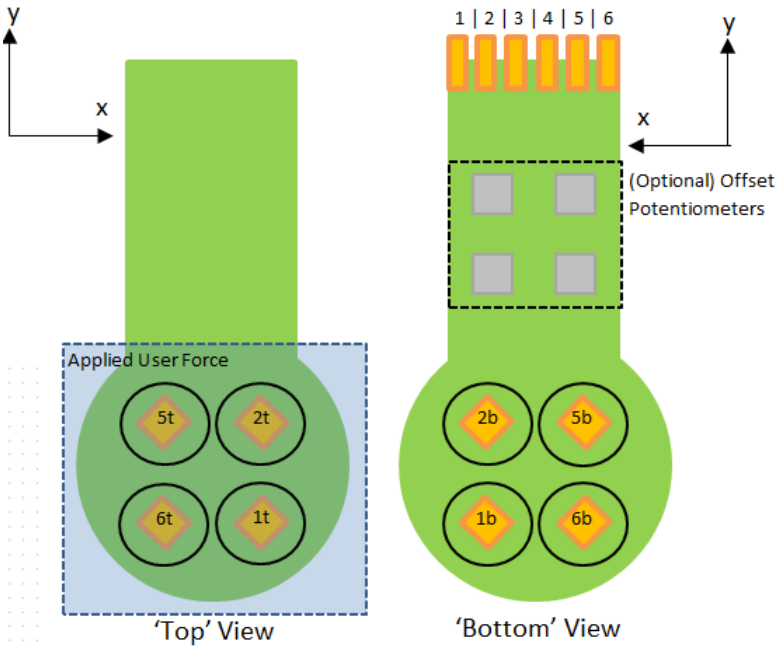


Figure 2.2: Force sensor component placement Force gages are labeled as yellow diamonds, offset potentiometers as gray squares, and FPC port connections as yellow rectangles.

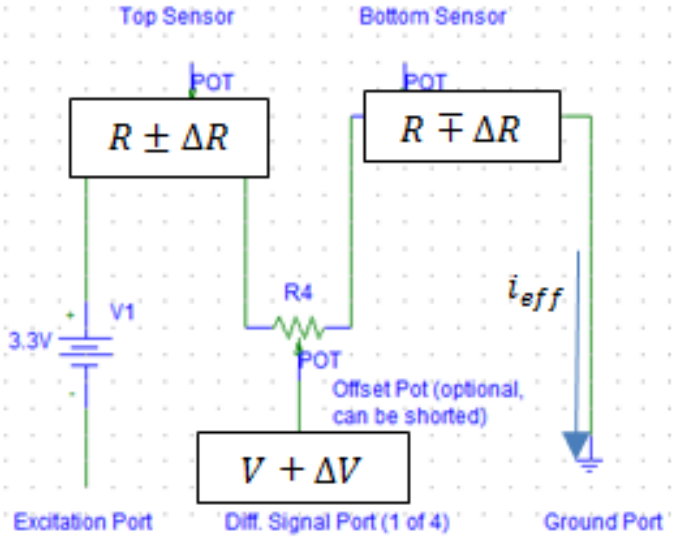


Figure 2.3: Voltage divider schematic A schematic of a single gage pair circuit shows the placement of the gages on either end of their shared potentiometer with the corresponding FPC signal port.

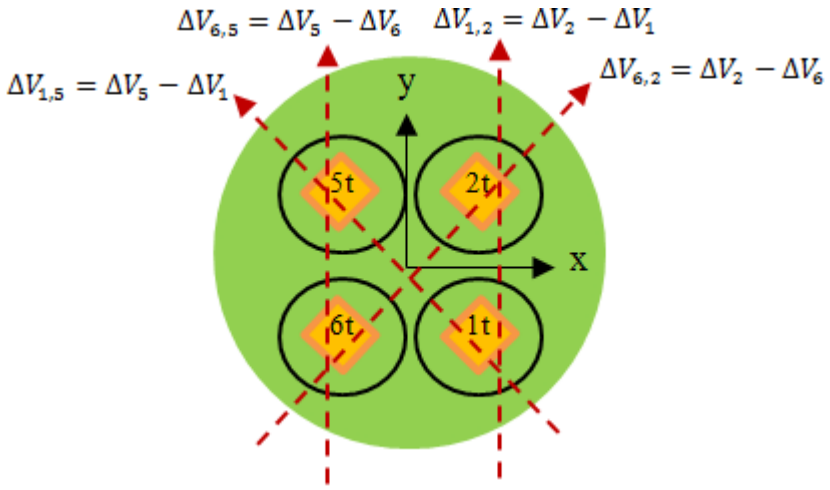


Figure 2.4: Voltage comparison between gages A top view of the force board with overlaid vectors representing the overall differences taken between individual voltage differences of gage pairs.

different Whetstone bridge circuits can be probed. From the current diamond configuration, four combinations results in four degrees of freedom of voltage deflection information. The specific probe combinations are shown by red lines on Fig. 2.4.

Based on the geometry of the board design, contributions to voltage differences from different force directions can be determined by the following equations:

$$V_x = \Delta V_{6,2} \cos\theta - \Delta V_{1,5} \sin\theta \tag{2.3}$$

$$V_y = \Delta V_{6,2} \sin\theta + \Delta V_{1,5} \cos\theta \tag{2.4}$$

$$V_z = \Delta V_{6,5} + \Delta V_{1,2} \tag{2.5}$$

$$e = |\Delta V_{6,2} - \Delta V_{1,5} + \Delta V_{6,5} - \Delta V_{1,2}| \tag{2.6}$$

where θ is the angle of the vector representing the probe difference from the x-axis and e is an error metric that is non-zero when non-ideal resistive changes occur on the board. Since the pattern is a square diamond, $\theta = 45^\circ$ so that Eqn. 2.3-2.6 become

$$V_x = \frac{\Delta V_{6,2} - \Delta V_{1,5}}{\sqrt{2}}$$

$$V_y = \frac{\Delta V_{6,2} + \Delta V_{1,5}}{\sqrt{2}}$$

$$V_z = \Delta V_{6,5} + \Delta V_{1,2}$$

$$e = |\Delta V_{6,2} - \Delta V_{1,5} + \Delta V_{6,5} - \Delta V_{1,2}|$$

The directional contributions can then be mapped to units of forces through a comprehensive calibration routine.

The force sensor boards are mounted underneath aluminum sensor beams, as shown in Fig. 2.5. In Fig. 2.6, the sensor board (labeled J) and sensor beam (labeled F) for a given finger are secured to the Hand Device by a pressure pad (labeled D) housed inside a sensor turret (labeled C) driven by a pressure screw (labeled B). A frame is secured to the top side of the PCB, shown in the top view of Fig. 2.1, and attached to a finger cup base with silicone cup (labeled H and G in Fig. 2.6). When a user fits their finger into the silicone cup and exerts force, the frame attachment scheme ensures the boards will deform significantly to produce the intended voltage differences.

With offset calibration now being handled through firmware, the offset potentiometers no longer have an essential purpose for the Hand Device and take up space. In the newer designs, the potentiometers are removed and

Figure 2.5: Fitted and blank views of force sensor A force sensor attached underneath a sensor beam (top) and a blank, unattached force sensor (bottom) laid next to a ruler scale.

boards are shortened as shown in Fig. 2.7 and 2.8. In addition, silkscreen has been removed from the bonding site between the strain gages and the substrate to allow for better adhesion and less material to absorb deflection, which should lead to stronger and less noisy voltage differences.

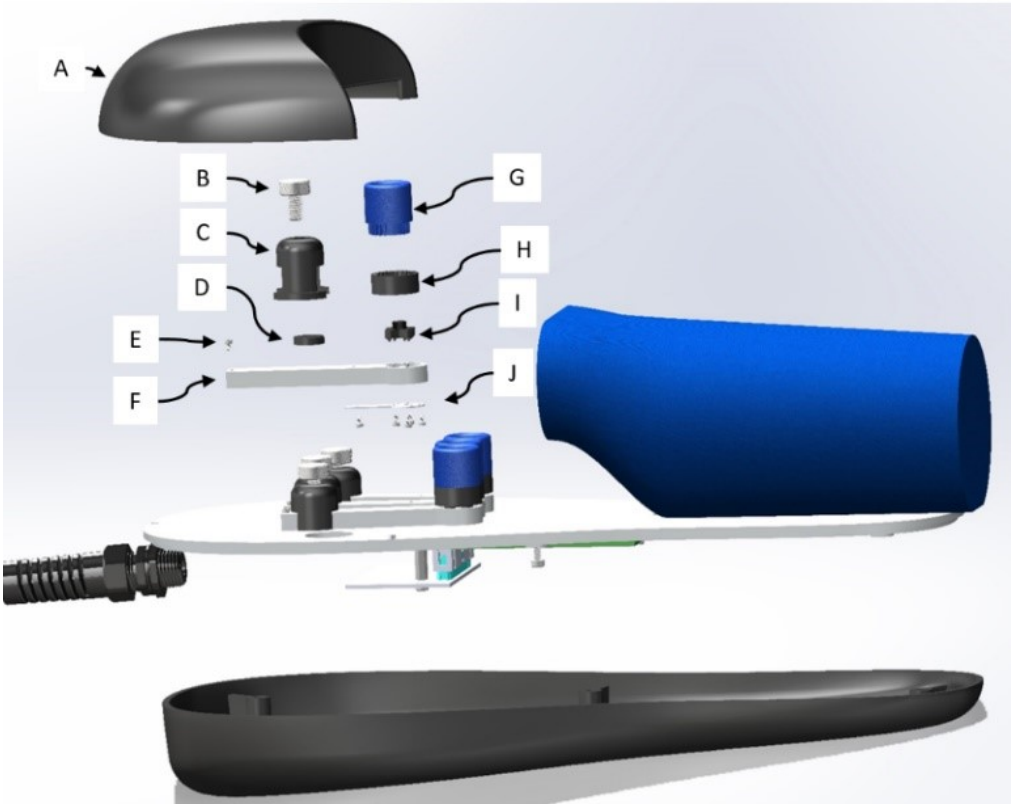
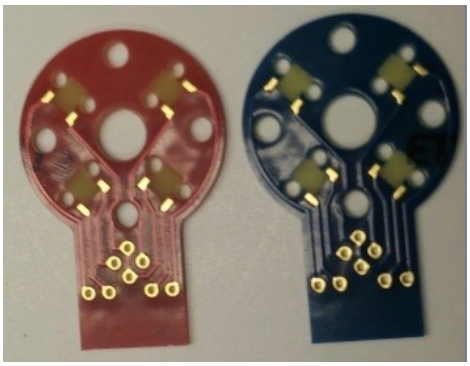
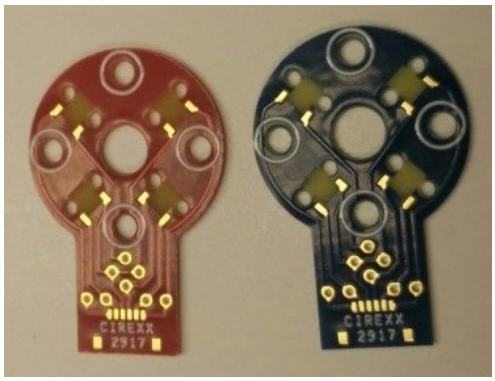


Figure 2.6: Exploded view of overall Hand Device The list of components labeled in the exploded view is as follows: chassis cover (A), pressure screw (B), sensor turret (C), pressure pad (D), retention screw (E), sensor beam (F), silicone finger cup (G), finger cup base (H), force sensor frame (I), and force sensor board (J).



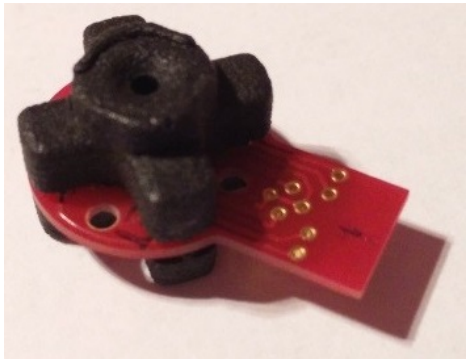
(a) Top views of 1.0 mm (left) and 0.5 mm (right) thick boards



(b) Bottom views of 1.0 mm (left) and 0.5 mm (right) thick boards

Figure 2.7: New force sensors (blank) New versions of finger boards feature a shorter length and the lack of offset potentiometers.

(a) Fitted lower force frame



(b) Fitted upper force frame

Figure 2.8: New force sensors with frame A shorter 0.5 mm thick board is fitted with a nylon attachment frame. (Note the removed potentiometers and shorter length.)

Chapter 3

Electrical Design

The custom data acquisition unit is a four-layer printed circuit board with three categories of surface-mount elements: the microprocessor, the amplifiers, and the multiplexers. The schematic of the DAQ unit is shown in Appendix A.

The microprocessor is a Teensy 3.5 USB Development board from PJRC, as shown in Fig. 3.1. The purpose of this board is to process and organize amplified signals from the finger sensors into readable serial information for the computer. It can also process commands from the computer to modify the amplifying transfer function and other characteristics for each amplifier through I^2C communication. The board has a footprint of 2.4 by 0.7 inches, which is smaller to the current NI DAQ board that is 3.9 by 2.5 inches. The microprocessor can collect analog signals on up to 23 ADC channels. At our design resolution of 16 bits, sampling has a theoretical upper limit of 12 MHz, which for twenty channels can mean up to 600 kHz per channel. The microcontroller is programmable through the standard Arduino IDE with the Teensyduino add-on, and has an internal RAM storage of 192 kB to permit

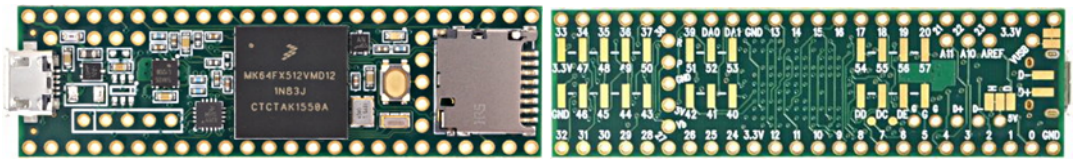
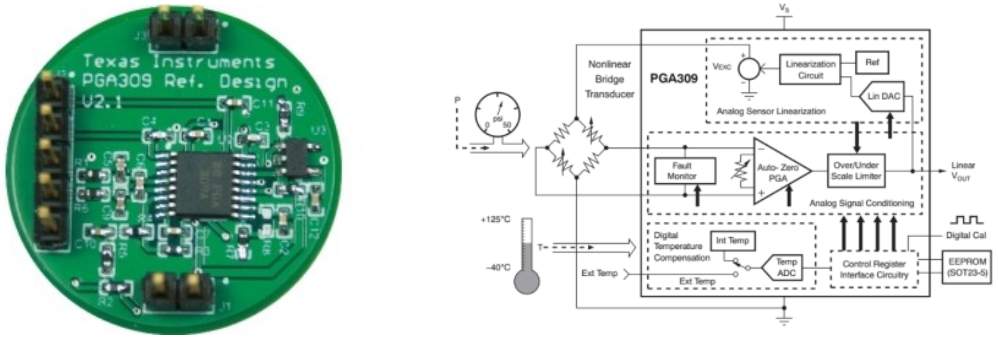


Figure 3.1: Teensy board views The front view (left) of the Teensy 3.5 USB board features a micro USB port, a processor, a micro SD adapter, and ports for soldering. The rear view (right) features additional contact pad ports for soldering.

storage of large arrays with digitized readings.

All amplifiers are of the PGA309 package from National Instruments, as shown in Fig. 3.2. The purpose of the amplifiers is to increase the sensitivity of the finger sensors. In general, the forces generated by the fingers of the patients induce voltage differences that are too small to analyze on the computer. However, device operators can adjust the gain function of the amplifiers, shown in Fig. 3.3, through I^2C commands to increase the magnitude and offset the voltage difference. The potential gain range of each amplifier is from 2.7 V/V to 1152 V/V, which is achieved through a three-stage amplifier system. The simplified schematic and sensitivity range details of the amplifier subsystem are shown in Fig. 3.3. More technical information is given in the PGA Datasheet and User's Guide online.

The multiplexers are of the TCA9548A package from National Instruments, as shown in Fig. 3.4. The purpose of the multiplexers is to provide address identifiers for each of the amplifiers so that communication from the microprocessor to a given amplifier is possible during calibration. By default, the PGA309 package has a fixed I^2C address of 0x40, and sending an I^2C command on the same data line to all amplifiers will change them all at the same time. Since one of the goals is to provide channel-specific changes to the gain



(a) Physical view **(b)** Simplified block diagram

Figure 3.2: PGA309 amplifier views The physical view displays the PGA309 chip on a development board. The simplified block diagram shows various subunits of the amplifier dealing with stage gains, linearization, digital temperature conversion, etc.

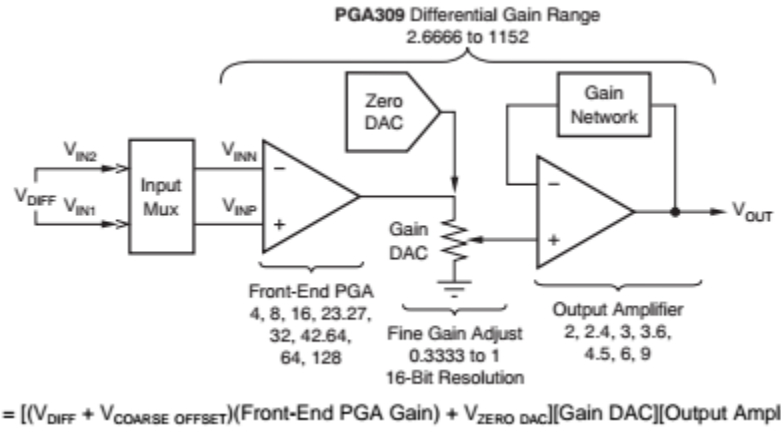


Figure 3.3: Gain schematic and function A schematic of the three-stage amplifier subsystem shows how signals are combined and amplified based on specified parameters. A gain transfer function is also shown that summarizes the signal process.

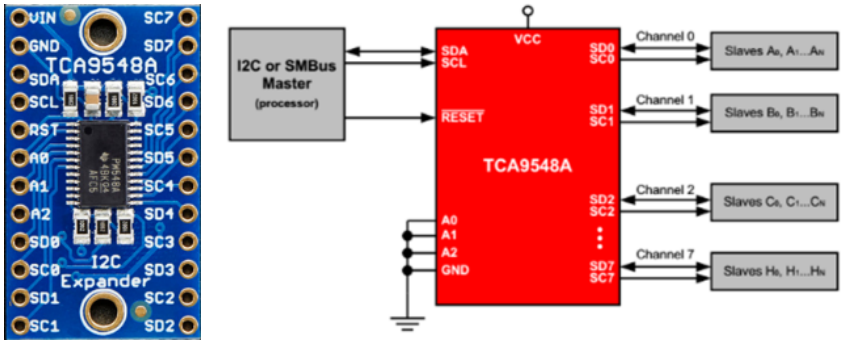


Figure 3.4: TCA9548A multiplexer The physical view depicts the TCA9548A multiplexer on a breakout board. The simplified block diagram shows how the TCA9548A multiplexer system interacts with master and slave devices.

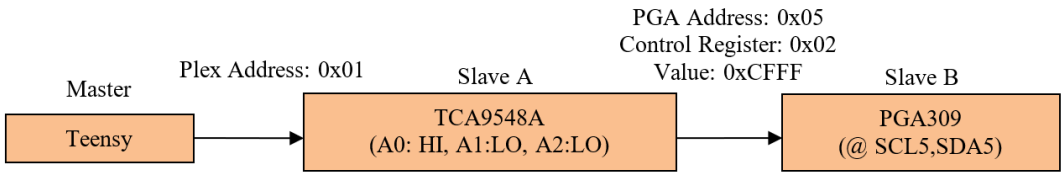


Figure 3.5: Multiplexer communication The process flow diagram depicts how the amplifier is accessed from Teensy through the multiplexer.

function, the amplifiers need to have addresses assigned from an external device. In this case, the TCA9548A assigns a unique address from 0x00 to 0x07 for up to eight I^2C-connected slave devices. Since there are a total of 20 slave amplifiers, the overall DAQ circuit needs a minimum of three multiplexers. The multiplexers themselves each have a hardware-adjustable I^2C address from 0x00 to 0x08. As a result, a master device (i.e. the Teensy) can access a single amplifier by sending the address of its corresponding multiplexer and the address that its multiplexer assigned it. A visual representation of the device access process is shown in Fig. 3.5. More information is given in the TCA95488 User's Guide online.

The following process flow for data collection is visualized in Fig. 3.6.

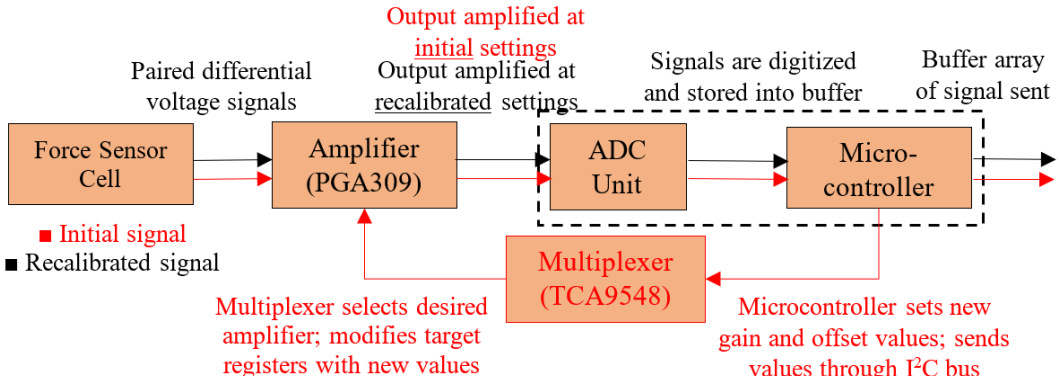


Figure 3.6: Overall startup communication This flow diagram describes how different physical components communicate during signal acquisition and recalibration.

First, the patient applies pressure to the finger sensor boards, which causes the strain gauges to flex. The difference in resistance causes a change in voltage for at least one signal channel. For the input of a given amplifier, two channels from the finger sensor are paired together to represent a component of force. The difference between the voltages of the two signals is amplified with initial gain and offset settings. The resulting amplified output signal is sent to the corresponding ADC channel of the Teensy. The voltage is converted to a 16-bit integer, with a reading of 65535 representing the analog reference and upper limit of 3.3 volts. The digitized readings are then stored into a buffer array before sending the packet over a USB connection to the computer. For the original prototype, sending buffer packets seemed more reliable than sending continuous stream data. However, later versions were able to send real-time data without sacrificing performance, as demonstrated later.

Readings from a particular amplifier can be used in a calibration routine to modify the transfer function of that amplifier to fit desired gain and offset parameters. First, the multiplexer and amplifier addresses are selected by the

Teensy for I^2C communication to prevent unnecessary writing to multiple amplifiers. Then, a control register address is selected corresponding to the particular characteristic setting that needs to be modified, described in Section 6 of the PGA309 User's Guide. The value at that address is rewritten to reflect the desired setting. In most cases, this should result in a modified transfer function and, in consequence, better sensitivity and offset.

With the circuit design validated in the breadboard prototype, a PCB version became feasible to implement. Printed circuit boards have cleaner and shorter wiring that will reduce noise, a smaller footprint that allows physical placement into the Hand Device, allows for efficient placement of all multiplexers and amplifiers within the DAQ subsystem, and is a much safer implementation for all users in terms of electrical exposure.

An intermediate version of the PCB is depicted in Fig. 3.7, while the final PCB design is shown in Fig. 3.8 and Appendix B. Designs were made in the Eagle PCB Design software, while actual manufacturing was handled by Advanced Circuits. The PCB is a four-layer mixed-signal design with trace wiring on the layer 1 (the upper layer) used mostly for sending 3.3 V power and reference, the digital ground pour plane on layer 2 (the layer underneath), the analog ground pour plane on layer 3, and mostly communication connections involving I^2C and raw analog readings on layer 4 (the bottom layer). The multiplexers, voltage regulator, and amplifiers are surface-mount (SMT) packages that can be soldered directly to pads on layer 1, whereas the Teensy board can interface with layer 1 through soldered header connections. Using header pin/socket connections made it easy to insert and remove the Teensy

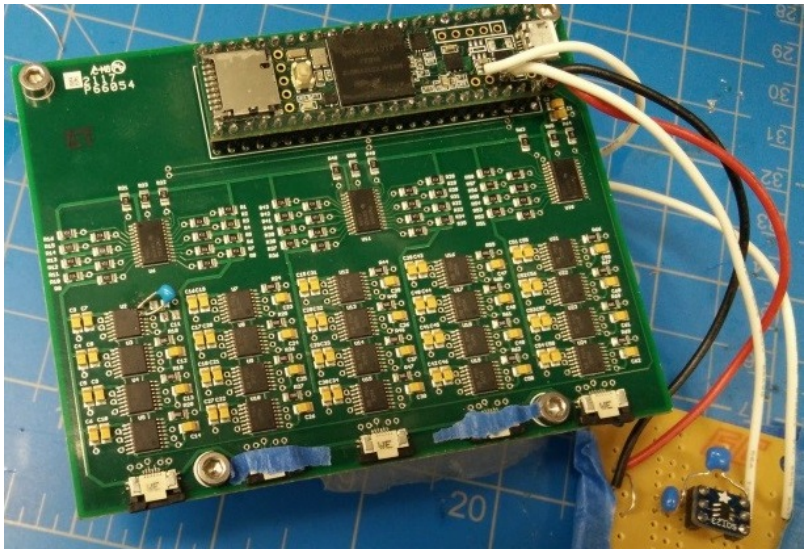


Figure 3.7: Old DAQ board An early prototype of the PCB board features the Teensy, multiplexers, amplifiers, FPC interfaces, and an external voltage regulator.

unit from the rest of the PCB in the intermediate version, but were ultimately replaced with permanent solder connections to decrease the board's thickness in the final version. The remaining capacitors, resistors, and finger interface pin connectors are also SMT components that can be soldered to layer 1.

The noise found in the validation phase should decrease from having large ground planes act as shielding for the PCB circuit, having shorter copper traces for wiring, and having external shielding on the inner wall of the Hand Device's mechanical housing.

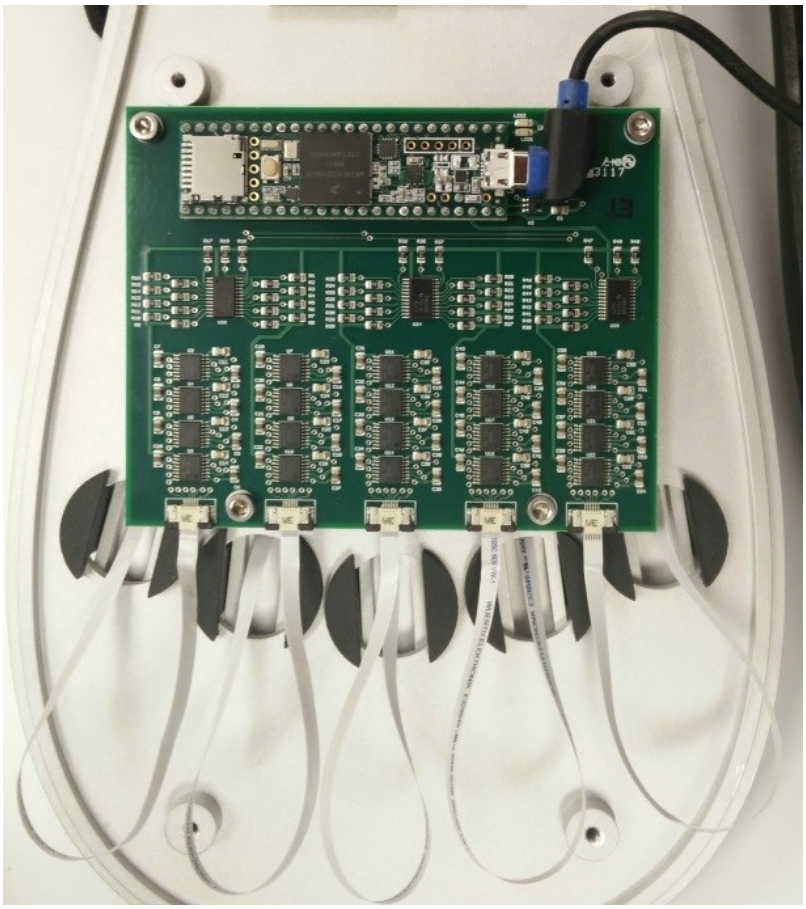


Figure 3.8: Final DAQ board A finalized PCB board is installed in the Hand Device and connected to a micro USB cable and to the finger beams underneath.

Chapter 4

API Design

The Hand Device transmits ADC readings and receives commands and power over a conventional micro USB cable connected into the Teensy, as shown in Fig. 3.8. To send appropriate I^2C commands to select PGA units of individual fingers upon user request, the Teensy was programmed with a HID messaging function that determines what registers of which PGA309 units to adjust based on the content of the user request. A typical state machine representation of device use is given in Fig. 4.1. Examples of commands are given in Fig. 4.2 while a full table of commands is given in Appendix C.

The calibration routine used in the API is as follows:

1. Records one second's worth of data for a given PGA/ADC channel.
2. Takes the average integer bit value, converts it to mV.
3. Queries the current stored gains for each amplifier stage from the last calibration (if the first calibration, just pulls default power-on settings).
4. Finds the estimated input PGA voltage into the PGA using known gains, the ADC reading, and the inverse transfer function.

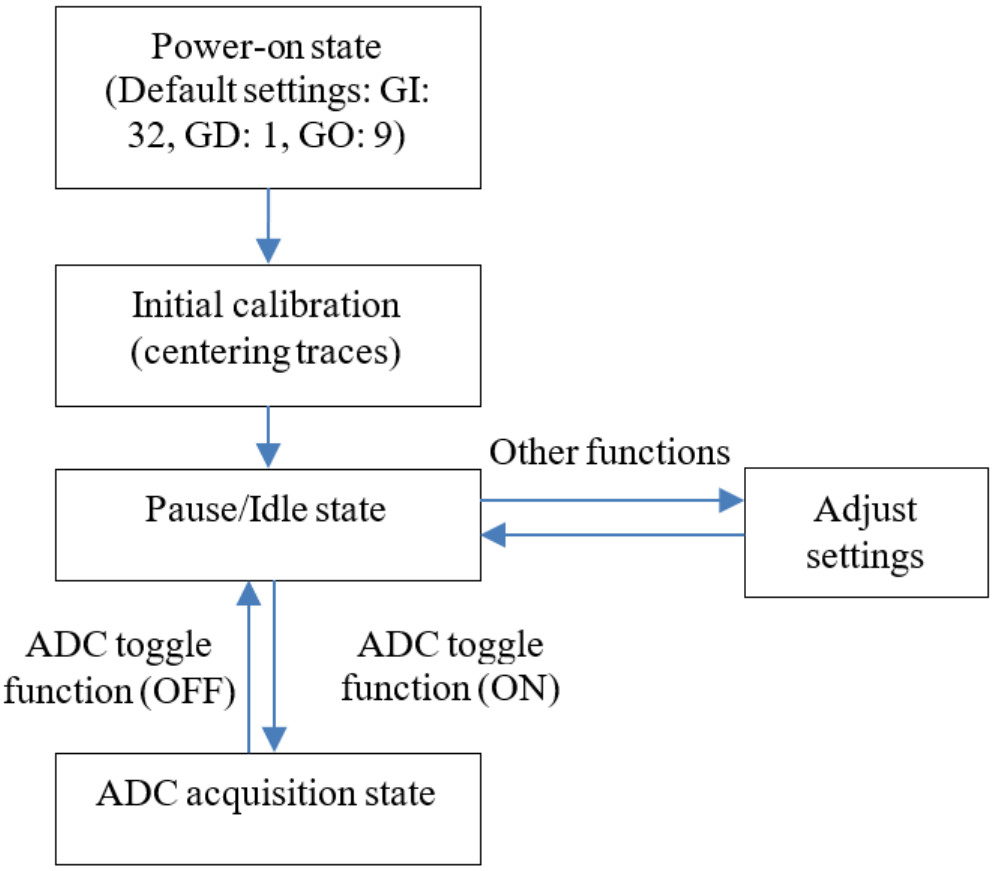


Figure 4.1: State machine This logic flow diagram describes the state machine behavior for typical Hand Device operation.

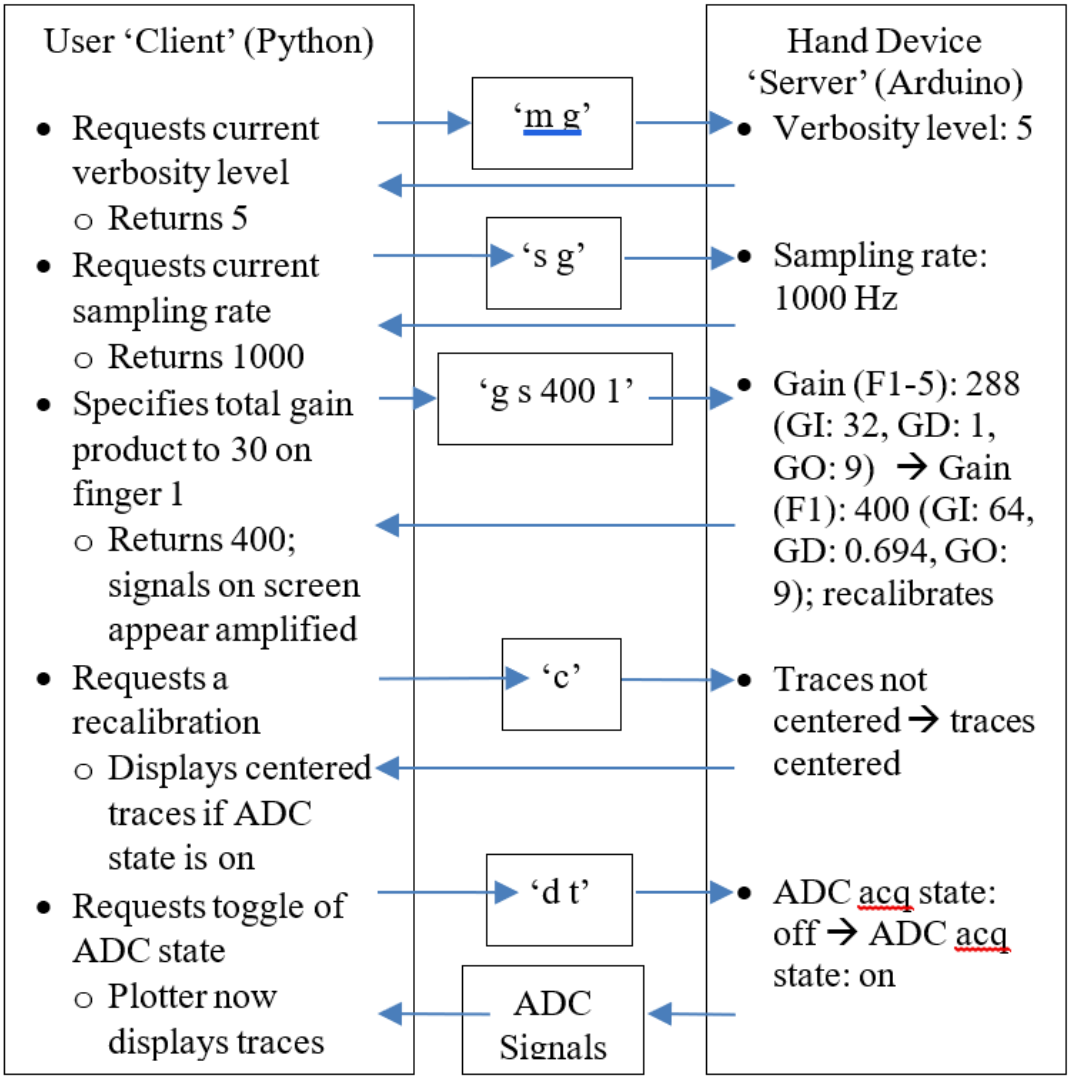


Figure 4.2: Python-Teensy communication Example command exchanges described between the client computer and the Hand Device.

5. Coarse offset V (with digitized levels as multiples of 0.85 * reference voltage) is set as closely to negative of estimated input to negate it.

6. If coarse offset is not equal to negative of estimated input, take the difference between the two to get the 'excess' voltage V.

7. Calculate fine offset V using the following equation:

$$V_F = \frac{f_{target}}{GD * GO} - \frac{GI * V_E}{1000 * V_{REF}} \tag{4.1}$$

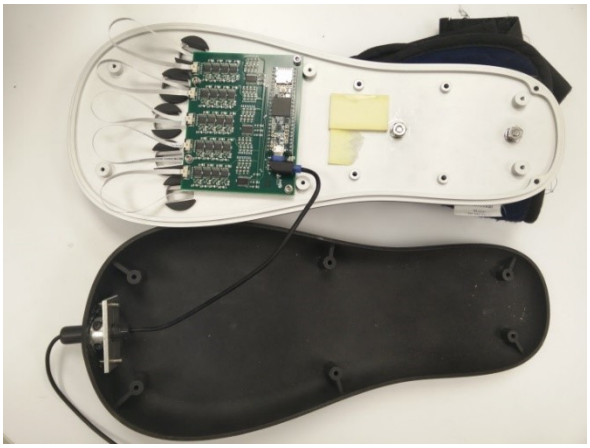
   where V_{REF} is the reference voltage in volts, V_E is the excitation voltage in millivolts, f_{target} is the intended fraction of reference voltage that an output reading should be given zero input (in this case being 0.5) and GI, GD, and GO are the existing gain settings for the input, fine, and output amplifier stages.

8. Rewrite the offset and gain registers to reflect the existing gain values and the new offset values.

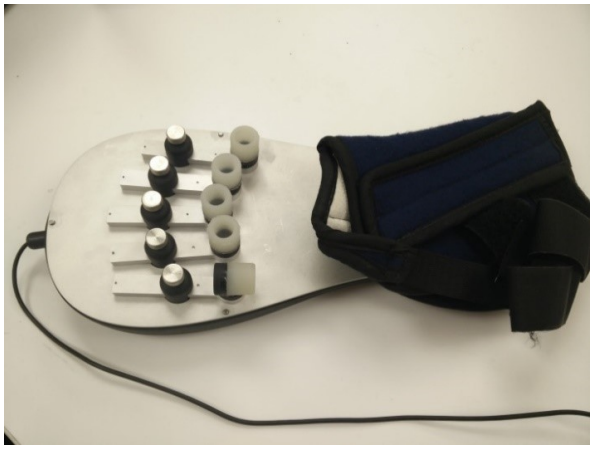
Chapter 5

System Integration

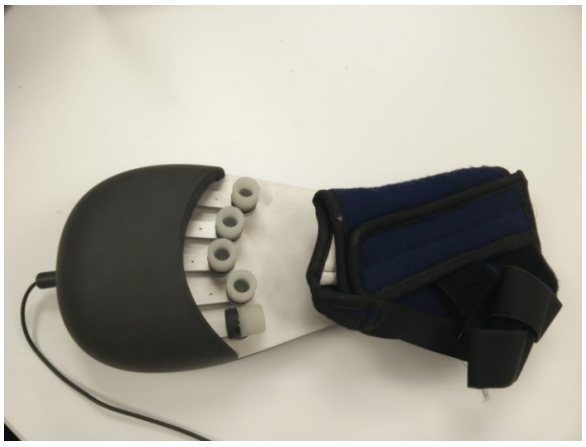
The most recent iteration of the Hand Device is assembled using laser-sintered nylon and 6061 aluminum for the structural chassis. The PCB is secured underneath an upper aluminum plate, as shown in the interior view of Fig. 5.1, with FPC ribbon cables running from each column of PGA amplifiers. The cables run through holes on the plate to the finger sensor units, secured to an adjustable aluminum beam specified as part F in Fig. 2.6. The upper plate also has a hand brace secured to stabilize the user's hand when fitting into the finger retention cups, shown in the upper views of Fig. 5.1. The upper plate is fastened to a lower nylon cover, which provides protection for the internal components. The USB connection cable runs from the Teensy on the PCB through the lower cover to a computer's USB port to provide HID communication.



(a) Interior view



(b) Exterior view without chassis cover



(c) Exterior view with cover

Figure 5.1: Hand Device views Various views are depicted of the Hand Device with disassembled, partially reassembled, and fully reassembled states.

Chapter 6

Experimental Validation

To ensure that the data acquisition board was going to function as expected when fabricated, a prototype was developed through a wired breadboard setup, as shown in Fig. 6.1. The breadboard features the Teensy microcontroller, two PGA309 amplifiers to switch I2C communication between, a TCA95488 multiplexer for assigning addresses, a 0.5 mm FPC pitch connector for finger board attachment, an alternative mock strain gauge with adjustable potentiometers, and multiple buttons for debug routines . A library of functions, tentatively named "pga-i2c", was developed for the Teensy to interface with the multiplexers and amplifiers.

Using the updated Teensy Wire and ADC libraries as dependencies, routines were developed for switching between multiplexer addresses, between multiplexer-assigned amplifier addresses, between read and write modes, and between the different control register addresses for accessing amplifier settings (nox771, 2018) (pedvide, 2017). Sample output for accessing and writing to amplifiers is given in Appendix G. Routines were also developed for collecting a stream of analog readings through up to 20 ADC channels and

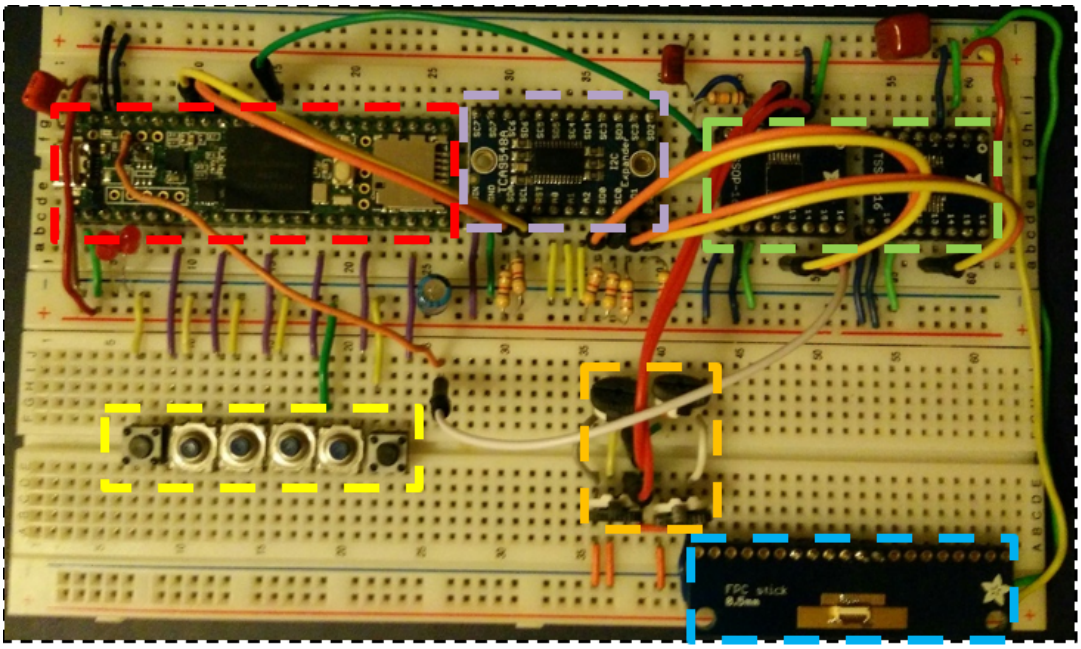


Figure 6.1: DAQ breadboard prototype List of the components in the breadboard prototype is as follows: Teensy microcontroller (red), multiplexer (purple), amplifiers (green), control buttons (yellow), voltage divider bridge (orange), and finger FPC connector (blue).

V_{ref}	$V_{offset,f}$	$V_{offset,c}$	GI	GD	GO	$V_{diff,abs}$	$V_{out,multi}$	$V_{out,ADC}$	$V_{out,calc}$	$\%error_{ADC}$
3.27	0.82	0	4	0.5	2	0.29	2	2.01	1.98	0.29
3.27	1.64	0	4	0.5	2	0.29	2.82	2.82	2.80	-0.03
3.27	0.82	0	4	0.33	2	0.29	1.33	1.33	1.31	0.17
3.27	0.82	0	4	1	2	0.11	2.49	2.50	2.52	0.39
3.27	0.82	-0.014	4	0.5	2	0.29	2	2.01	1.92	0.29
3.27	0.82	0	8	0.5	2	0.26	2.75	2.75	2.90	0.16
3.27	0.82	0	4	0.5	3	0.26	2.68	2.68	2.79	0.16
3.27	0.82	0	16	0.5	2	0.07	1.89	1.90	1.94	0.32
3.27	0.82	0	4	0.5	2.4	0.07	1.33	1.31	1.32	-1.71

Table 6.1: Amplifier verification summary A tabular summary of results for the amplifier setting change test with ADC and transfer function errors compared to multimeter readings.

Averaging Samples	Sampling Speed	Conversion Speed	Sampling Rate (kHz)	Averaging Noise
8	Medium	Medium	26.46	30
16	Medium	Medium	14.23	25
16	High	High	28.29	29
32	High	High	21.34	35

Table 6.2: Noise and sampling rate summary A summary of trials conducted to quantify noise and sampling rate under different speed and averaging settings.

for sending a buffer of readings over USB.

To filter out high frequency noise above 1000 Hz, a passive low-pass RC filter with a cutoff frequency of 100 Hz was wired using a 100nF capacitor and a 16kΩ resistor. It should be noted that this filter design was only a simple implementation to filter noise above the Nyquist frequency; a better filter implementation is utilized in the PCB version. The output was sent back to the feedback port of the amplifier for proper amplifier operation and for signal stability.

After selecting the appropriate resistors for I^2C communication and the appropriate capacitors for decoupling and filtering, a validation test was done

using the breadboard mockup to verify that changes to the PGA309 settings were reflected in modified output analog signals. Table 6.1 and Appendix D together state the setting changes for each trial, which can include gains for all three internal amplifiers (GI, GD, GO), and ratios for both coarse and fine offset. These settings are explained in the aforementioned Section 6 of the PGA309 User's Guide along with register addresses. Table 6.1 and Appendix D together also include the input signal voltage, the output voltage before and after digitization, and the output voltage predicted by the transfer function. The ADC did not appear to affect the voltage of the raw analog signal, since the error was at most 0.39% except for the last trial with an error of 1.71%. The transfer function also predicts the raw analog signal with a maximum error margin of 5.36% across all nine trials.

Afterwards, a performance test was conducted to gauge the sampling speed of the microcontroller. This involved modifying the ADC settings provided with the ADC library. Sampling and conversion speed were each changed through generic terms (i.e. "very low", "low", "medium", "high", "very high"). Output resolution and averaging interval were changed through numeric settings. The actual reading size was still 16 bits per sample, so packet size could not be reduced in size (pedvide, 2017). The results are summarized in Table 6.2 and shown in full on Appendix E. For the purposes of minimizing noise while keeping the sampling rate over at least 1 kHz per channel, the maximum averaging interval and resolution of 16 each should be selected with both speed settings set to high. (For 16-bit resolution, the performance of "very high" speed for both speed settings is equivalent to "high" speed.)

Within the same test, another observation was done to see the maximum noise being generated for an analog reading when no force input was applied to sensors. The target noiseless resolution for a given analog reading is at least 12 bits out of the 16 available bits. This is equivalent to having a maximum noise margin of 16 on the digitized reading. A sample ADC reading with inherent noise is given in Fig. 6.2. Noise was approximated by estimating the largest magnitude between an adjacent peak-trough pair in the signal. Due to limitations of the Serial Plotter interface on the Arduino IDE, only the last 500 samples of a continuous data stream are plotted at a given time, and data could not be exported from the Serial Plotter for analysis. Therefore, evaluation of noise could be conducted only on the last 500 samples. Based on findings summarized in Table 6.2, it seems that the breadboard prototype yields 5 bits of noise and can only provide up to 11 bits of noiseless data. The associated noise graphs to the tabular results are given in Appendix F. Part of the noise could be attributed to internal error inside the ADC unit(s) or from the long wiring used to operate the breadboard circuit; the analog reading waveforms suggest that EM noise from external sources did not have a significant contribution, although it has not completely been ruled out.

In order to test the new PCB and finger designs, a Python interface for a client PC known as plotosaurus has been developed to plot individual ADC readings as well as positional information of the fingers based on Eqns 2.3-2.6 (Forrence, 2017). The verifying protocol was relatively simple in that the data connection and power would be enabled for the Hand Device by plugging it in. After its initial calibration routine, the plotosaurus app would be launched

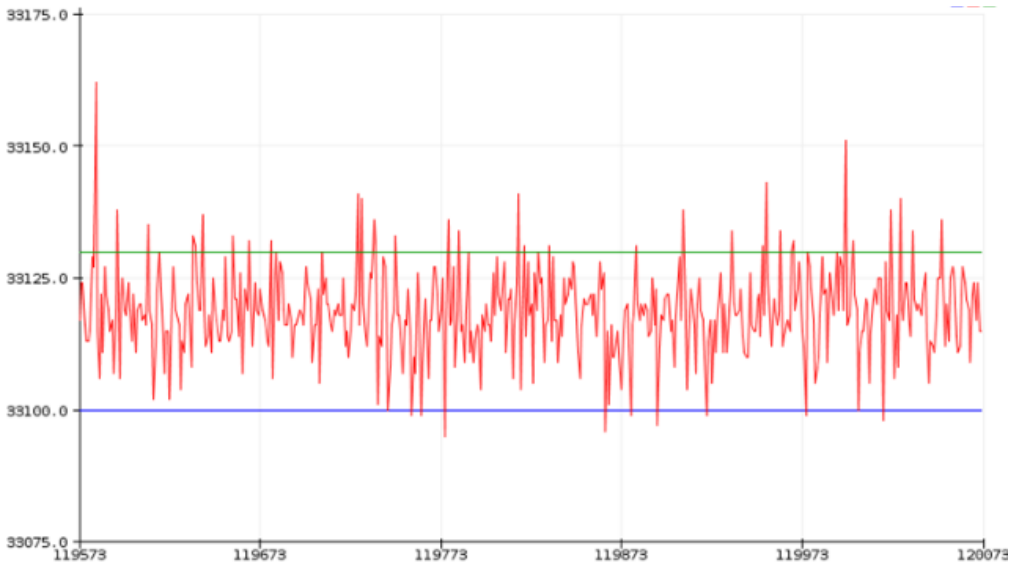
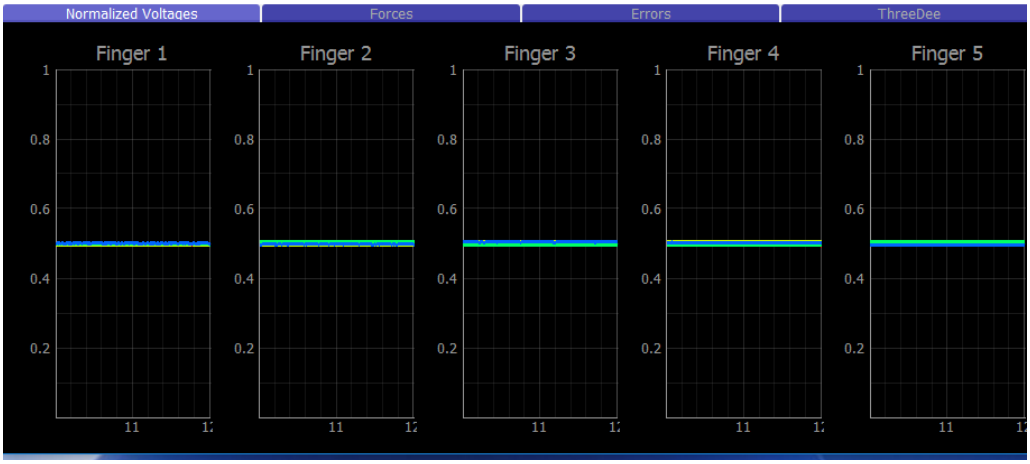


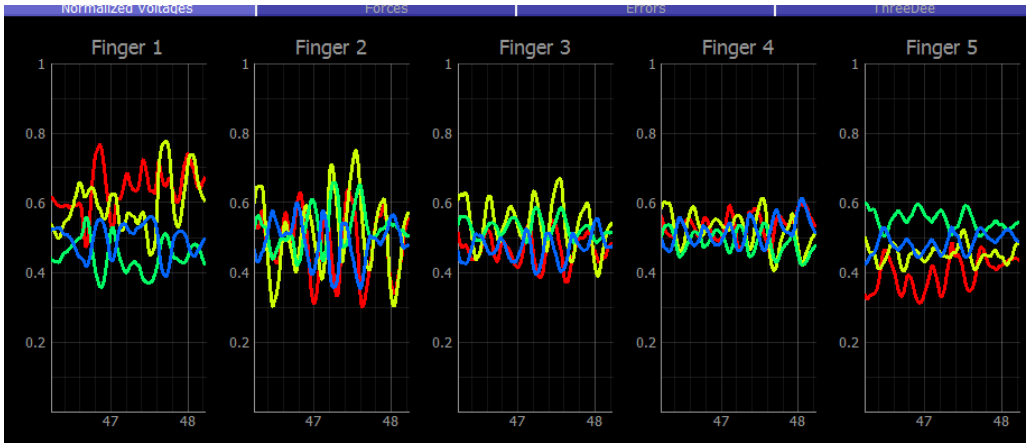
Figure 6.2: Sample ADC recording for noise Digitized 16-bit voltage readings are shown in red for the last 500 samples of the third trial. Lines used to control upper and lower axis limits for voltage are in green and blue. (Settings: 16-sample averaging, high sampling speed, high conversion speed)

on the voltage reading screen to see if the voltage readings have been centered. After confirming, all fingers of the Device would have persistent force excitation for several seconds to see if all channels are recording deflections. Finally, the finger boards would be returned to rest to see whether hysteresis would be an issue or not. From the results in Fig. 6.3, it is apparent that the Device does indeed center the voltage readings to the desired fractional level of 0.5, exhibit reasonable changes in force readings when under stimulation and displayed on screen, and demonstrates acceptable hysteresis and repeatability behavior (within 2-3% of reference voltage). Therefore, it appears that the new revision is functioning as intended.

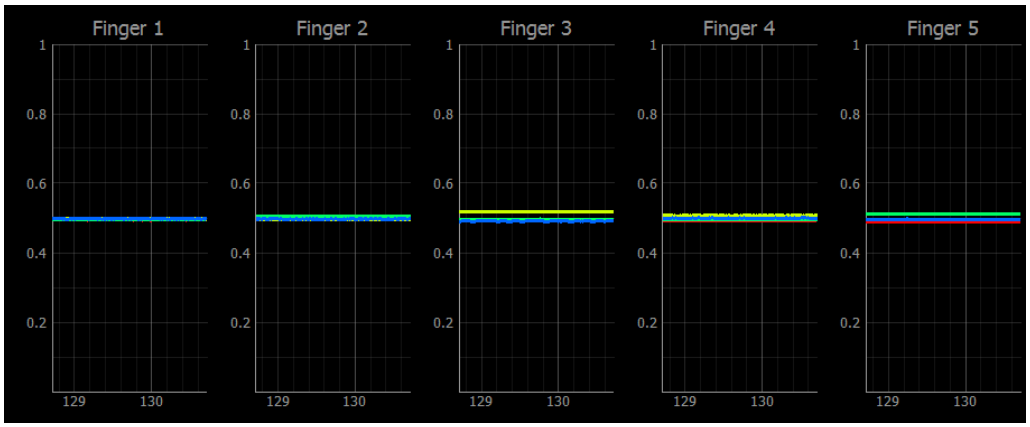
To determine the viability of mapping force units to relative voltage outputs in the future, a Wagner Force Ten digital force gage is fitted into a given



(a) Before stimulus



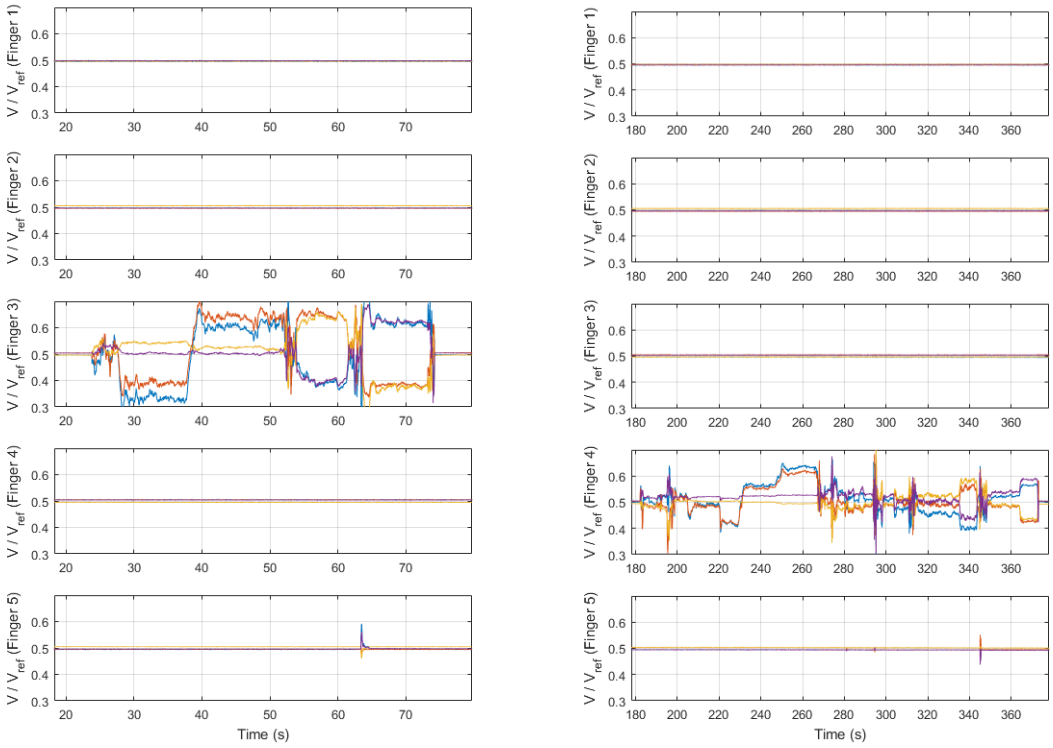
(b) During persistent stimuli



(c) Traces after stimuli

Figure 6.3: Hand Device recordings via HID Normalized ADC readings from HID communication are printed to the plotosaurus voltage display for certain stimulus conditions.

finger and held at various constant levels of force. Meanwhile, the plotosaurus tool logs all voltage samples from the force trials into CSV files. Using the graph brush tool in MATLAB on plots of these imported files, the constant force levels were isolated for data analysis by locating the earliest point of the voltage 'plateau' to the latest point. The results are summarized in Fig. 6.4 and Table 6.3. Brush selection graphs are depicted in Appendix H. From the reproducible patterns for certain forces, the prospect of force calibration seems promising.



(a) Third finger sensor with 1 N force exerted in four axial directions on the XY plane

(b) Fourth finger sensor with 0.1 N force exerted followed by 1 N force in all four directions

Figure 6.4: Fixed-force voltage data Sample trials of the voltage traces generated at set force levels are plotted in MATLAB and can be analyzed for voltage-force calibration.

Timestamp (min:sec)	Fig	Digit	Time Start (sec)	Time End (sec)	Sample Count	Ch1 Avg (V)	Ch2 Avg (V)	Ch3 Avg (V)	Ch4 Avg (V)	Ch1 Std (V)	Ch2 Std (V)	Ch3 Std (V)	Ch4 Std (V)	Force (N)	Axis
14:15.1	a	3	18.305	23.687	5068	0.4963	0.5047	0.496	0.5038	0.0004	0.0005	0.0003	0.0004	0.0	–
14:15.1	a	3	28.155	37.431	8610	0.3324	0.3897	0.5433	0.5018	0.0113	0.0093	0.0031	0.0025	-1.0	Y
14:15.1	a	3	40.295	51.404	10988	0.6025	0.6393	0.5242	0.5019	0.0153	0.014	0.0042	0.0029	1.0	Y
14:15.1	a	3	54.383	61.253	6803	0.3954	0.6493	0.6371	0.3988	0.0168	0.0103	0.0106	0.0092	-1.0	X
14:15.1	a	3	65.149	72.936	7725	0.6189	0.3845	0.3742	0.6233	0.0113	0.0049	0.0089	0.0053	1.0	X
14:53.6	b	4	208.719	219.508	9639	0.4902	0.4851	0.5029	0.5209	0.0046	0.0037	0.0014	0.0011	-0.1	Y
14:53.6	b	4	221.249	228.558	6685	0.4206	0.423	0.5044	0.5149	0.0067	0.0053	0.0009	0.0012	-1.0	Y
14:53.5	b	4	233.083	246.308	12900	0.5644	0.5543	0.5006	0.5238	0.0038	0.0037	0.0007	0.0007	0.1	Y
14:53.6	b	4	251.182	265.103	13365	0.632	0.6126	0.4944	0.5269	0.005	0.0045	0.0008	0.0007	1.0	Y
14:53.6	b	4	315.921	334.018	17003	0.4562	0.4918	0.5246	0.5033	0.0065	0.0062	0.0053	0.0056	0.1	X
14:53.5	b	4	336.467	342.024	5389	0.401	0.5563	0.582	0.4402	0.0045	0.0076	0.0054	0.0059	1.0	X
14:53.6	b	4	350.378	362.559	11730	0.5217	0.4847	0.4883	0.5382	0.0036	0.0038	0.0034	0.0033	-0.1	X
14:53.6	b	4	365.344	372.305	6619	0.5648	0.4265	0.436	0.5868	0.0021	0.0024	0.0022	0.002	-1.0	X

Table 6.3: Finger channel voltage statistics A tabular summary of results describe central tendency and spread of voltages at various constant-force stimulus intervals. These statistics are based on data depicted in Figure 6.4.

Chapter 7

Discussion and Conclusion

The aforementioned custom DAQ board provides the necessary amount of independent analog channels needed to sense three-dimensional forces from the user's fingers fitted into the Stroke Rehabilitation Hand Device. From the validation tests, modifying the gain function works as intended and the noise is manageable. The basic DAQ design is validated on breadboard and PCB in terms of functionality and noise/artifact management. From here, force unit calibration can proceed so that proper units of force measurement (e.g. Newtons, pounds, etc.) can be mapped to voltage levels depending on finger, gain levels, and force direction. Once multiple units are fully assembled and undergo fatigue testing, the Hand Device is planned to undergo holistic validation for healthy and chronic patients at Western University in Canada, and for acute patients within the Johns Hopkins Department of Neurology. If said validation passes, the Stroke Rehabilitation Hand Device should be a suitable candidate for tracking and guiding the improvement of hand control of stroke patients. This in turn should lead to better recovery in both the short and long term and result in a higher quality of life for patients.

References

E. Cruz, H. Waldinger and D. Kamper (2005). "Kinetic and kinematic workspaces of the index finger following stroke." In: *Brain*.

S. Li M. Latash, G. Yue V. Siemionow V. Sahgal (2003). "The effects of stroke and age on finger interaction in multi-finger force production tasks." In: *Clinical Neurophysiology* 114.9, pp. 1646–1655.

A. Fugl-Meyer L. Jääsko, I. Leyman S. Olsson and S. Steglind (1975). "The post-stroke hemiplegic patient. 1. a method for evaluation of physical performance." In: *International Journal of Rehabilitation Research* 7.1, pp. 13–31.

Lyle, R. (1981). "A performance test for assessment of upper limb function in physical rehabilitation treatment and research." In: *Scandinavian Journal of Rehabilitation Medicine* 4.4, pp. 483–492.

J. Xu N. Ejaz, B. Hertler M. Branscheildt M. Widmer A. Faria M. Harran J. Cortes N. Kim P. Celnik T. Kitago A. Luft J. Krakauer and J. Diedrichsen (2016). "Recovery of hand function after stroke: separable systems for finger strength and control". In: *bioRxiv*.

J. Metzger O. Lambercy, A. Califfi D. Dinacci C. Petrillo P. Rossi F. Conti and R. Gassert (2014). "Assessment-driven selection and adaptation of exercise difficulty in robot-assisted therapy: a pilot study with a hand rehabilitation robot." In: *Journal of NeuroEngineering and Rehabilitation*.

J. Metzger O. Lambercy, D. Chapuis and R. Gassert (2011). "Design and characterization of the ReHapticKnob, a robot for assessment and therapy of hand function." In: *Intelligent Robots and Systems*.

S. Ito H. Kawasaki, Y. Ishigure Y. Nishimoto T. Aoki T. Mouri H. Sakaeda and M. Abe (2007). "Development of a Hand Motion Assist Robot for Rehabilitation Therapy by Patient Self-Motion Control". In: *Proceedings of the 2007 IEEE 10th International Conference on Rehabilitation Robotics*.

L. Dovat O. Lambercy, R. Gassert T. Maeder T. Milner T. Leong and E. Burdet (2008). "HandCARE: a cable-actuated rehabilitation system to train

hand function after stroke." In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering*.

Tyromotion (2017). *AMADEO*. URL: http://tyromotion.com/en/products/amadeo (visited on 02/10/2018).

NEOFECT (2016). *RAPAEL Smart Glove | NEOFECT*. URL: http://www.neofect.com/en/product/rapael/ (visited on 02/10/2018).

nox771 (2018). *i2c_t3*. URL: https://github.com/nox771/i2c_t3 (visited on 02/10/2018).

pedvide (2017). *ADC*. URL: https://github.com/pedvide/ADC (visited on 02/10/2018).

Forrence, Alex (2017). *plotosaurus*. URL: https://gitlab.com/aforren1/plotosaurus (visited on 02/10/2018).

Appendices

Appendix A

Board Schematic

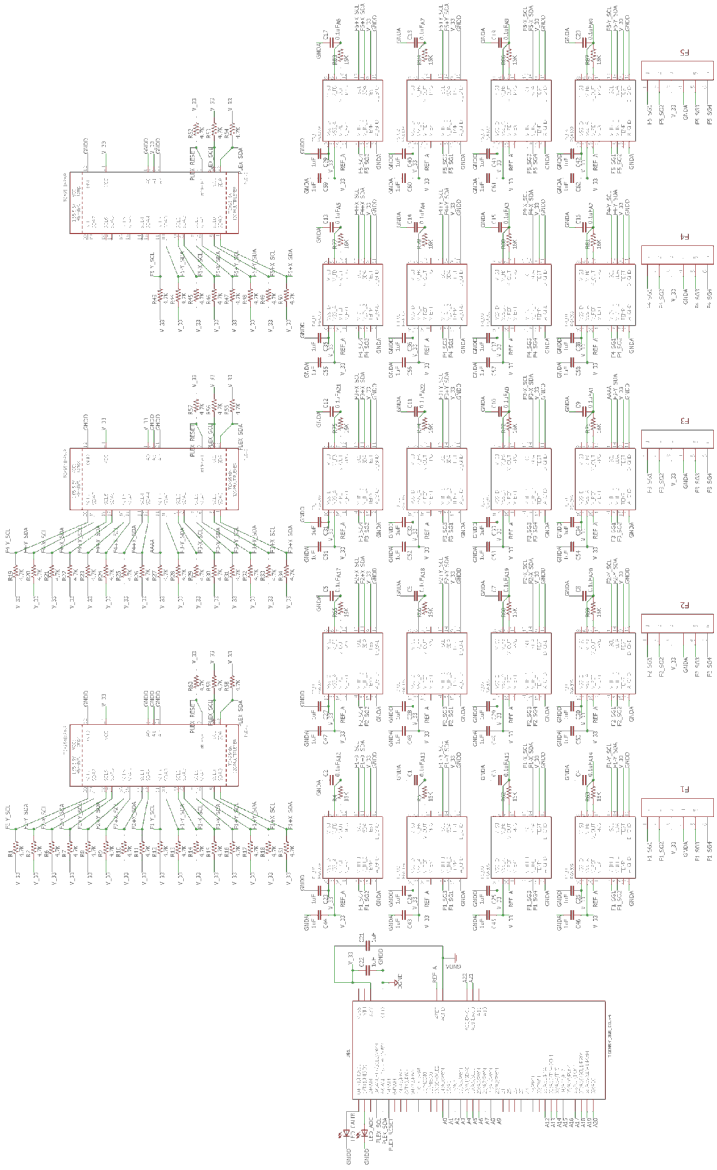


Figure A.1: A schematic of the DAQ board shown in the Eagle PCB design software.

Appendix B

Board Trace and Component Placement Views

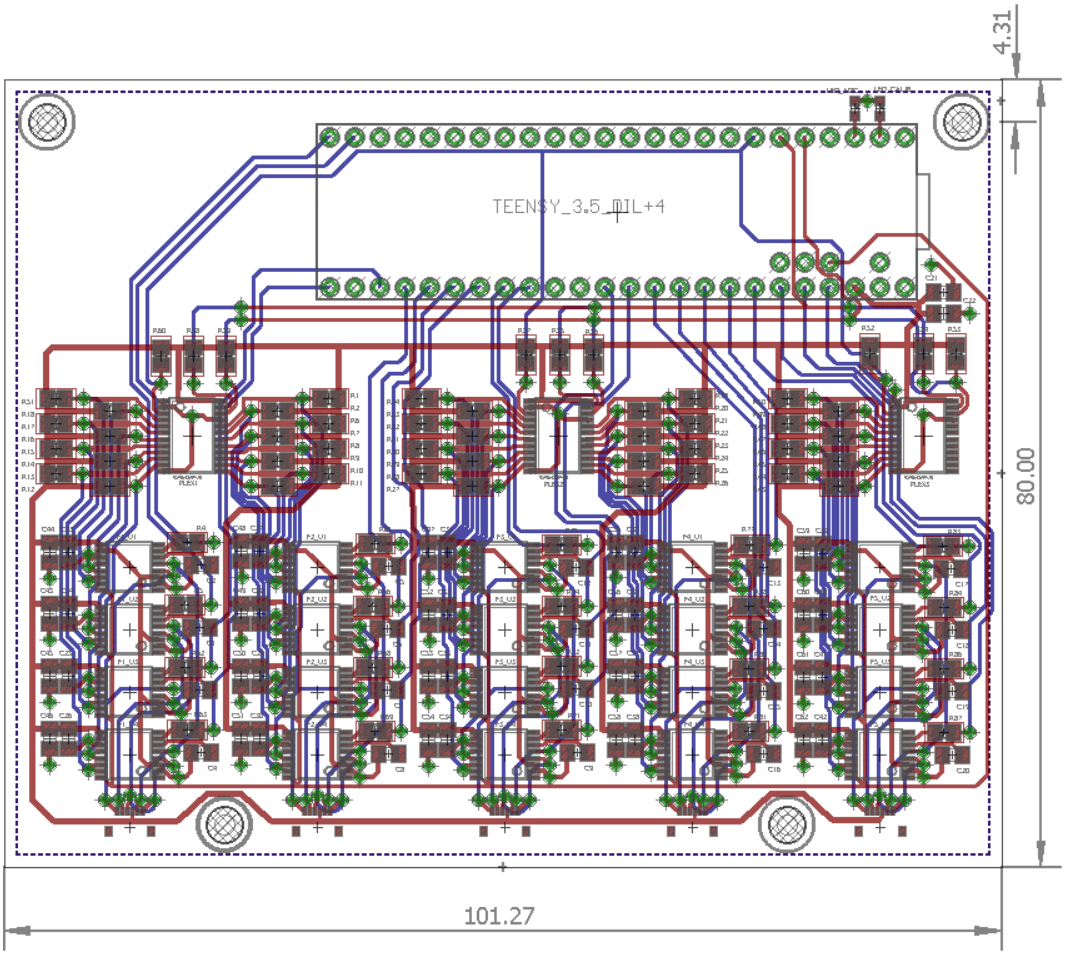


Figure B.1: An overall view of the PCB depicits trace layers and components, but hides the ground layers.

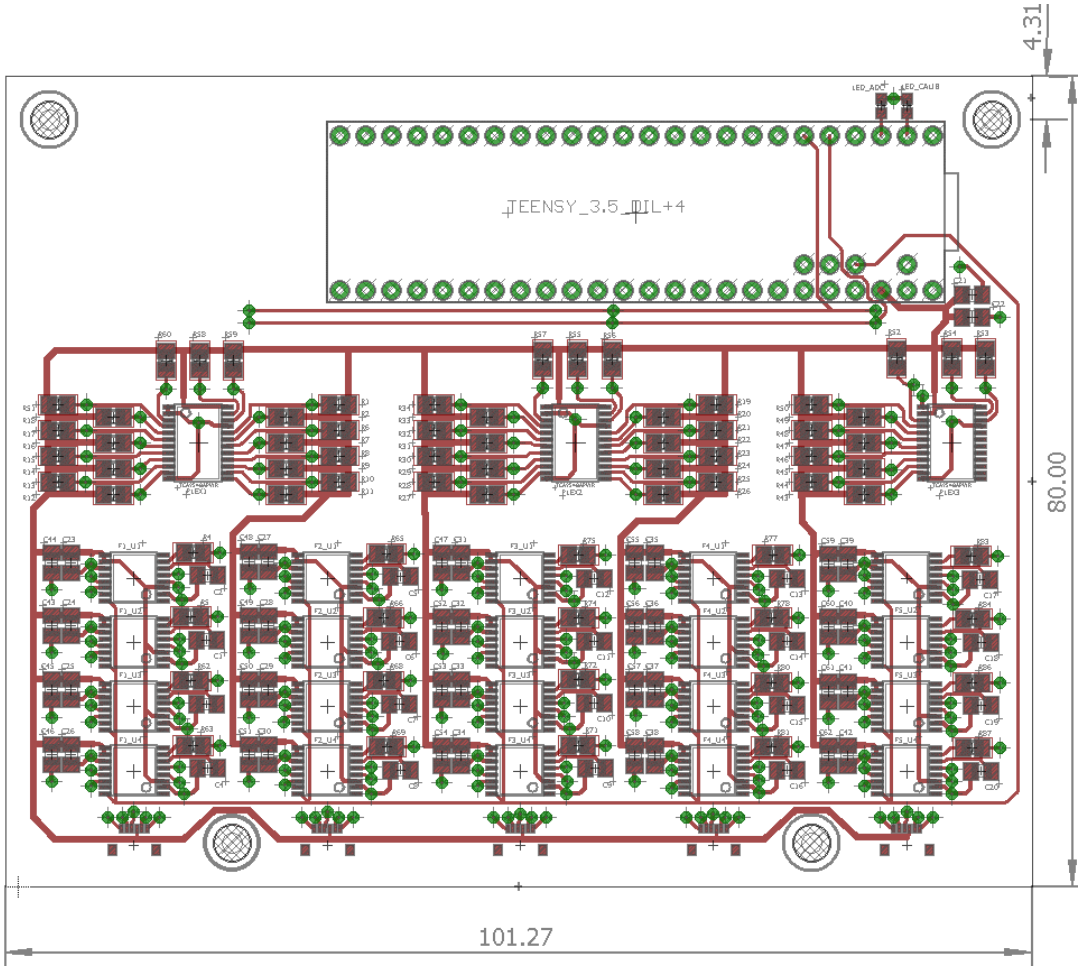


Figure B.2: The top trace layer view of the PCB is shown only.

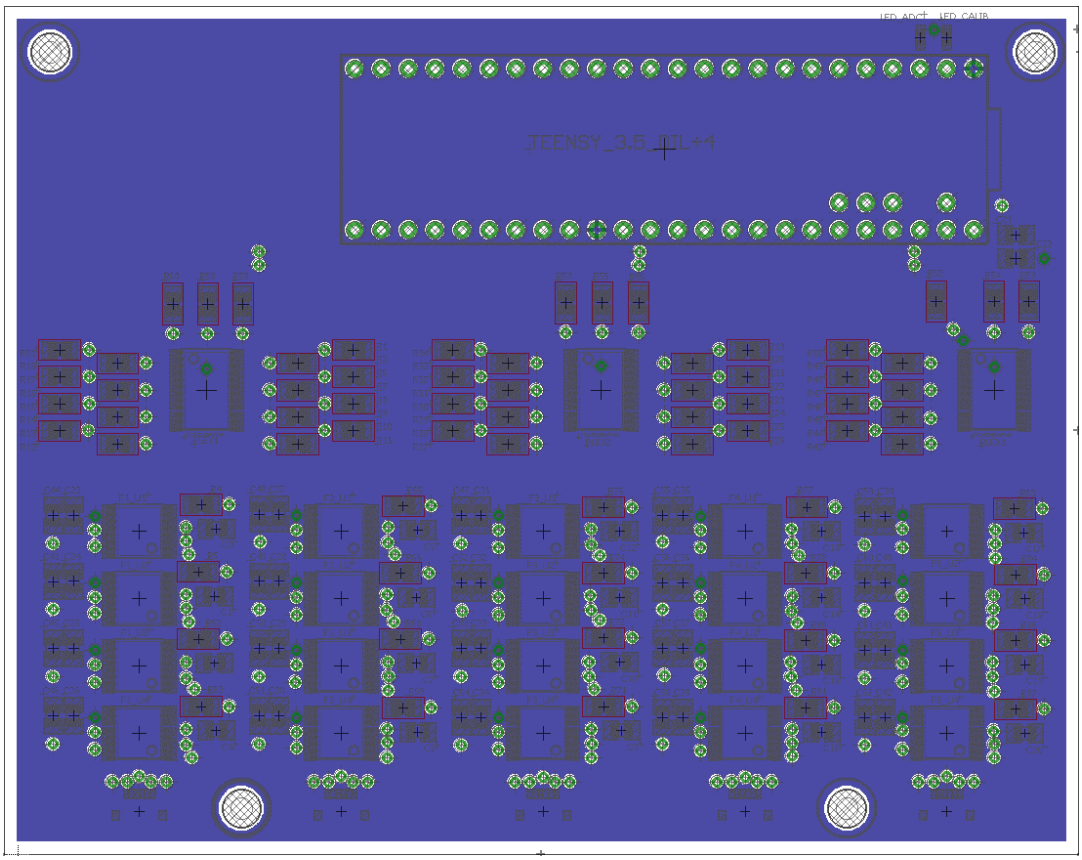


Figure B.3: The digital ground layer view of the PCB is shown only.

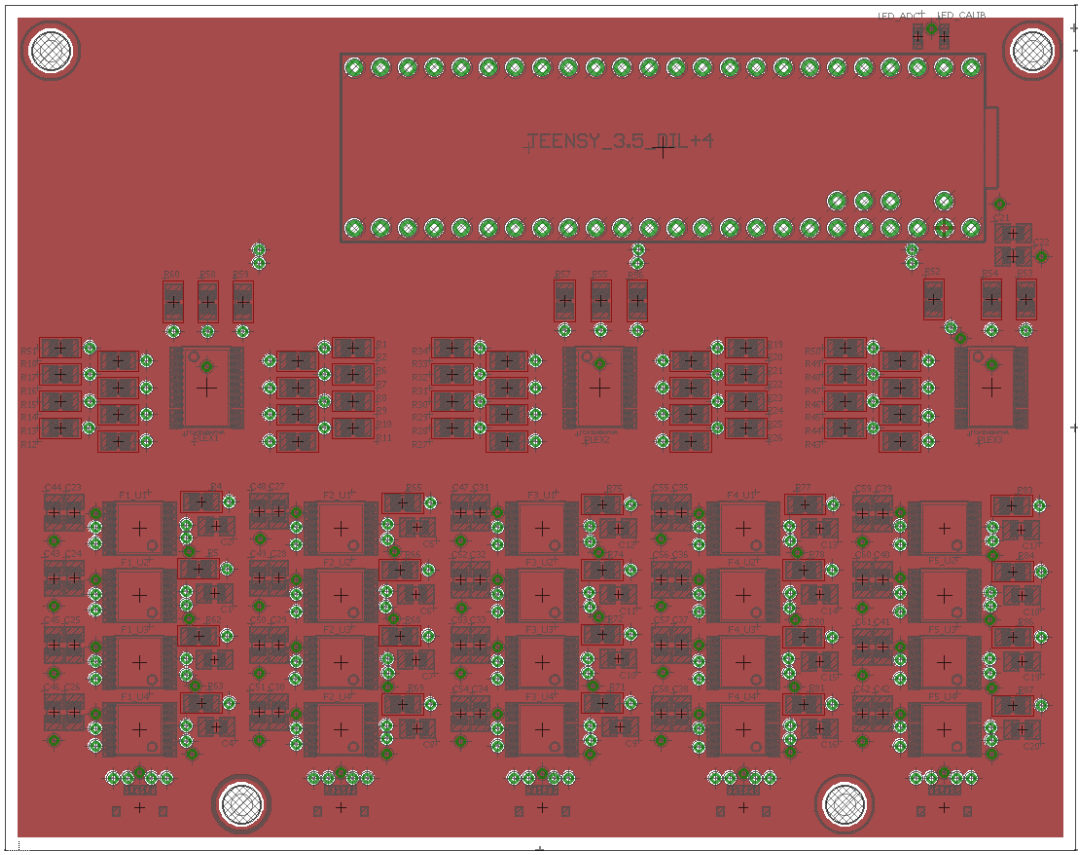


Figure B.4: The analog ground layer view of the PCB is shown only.

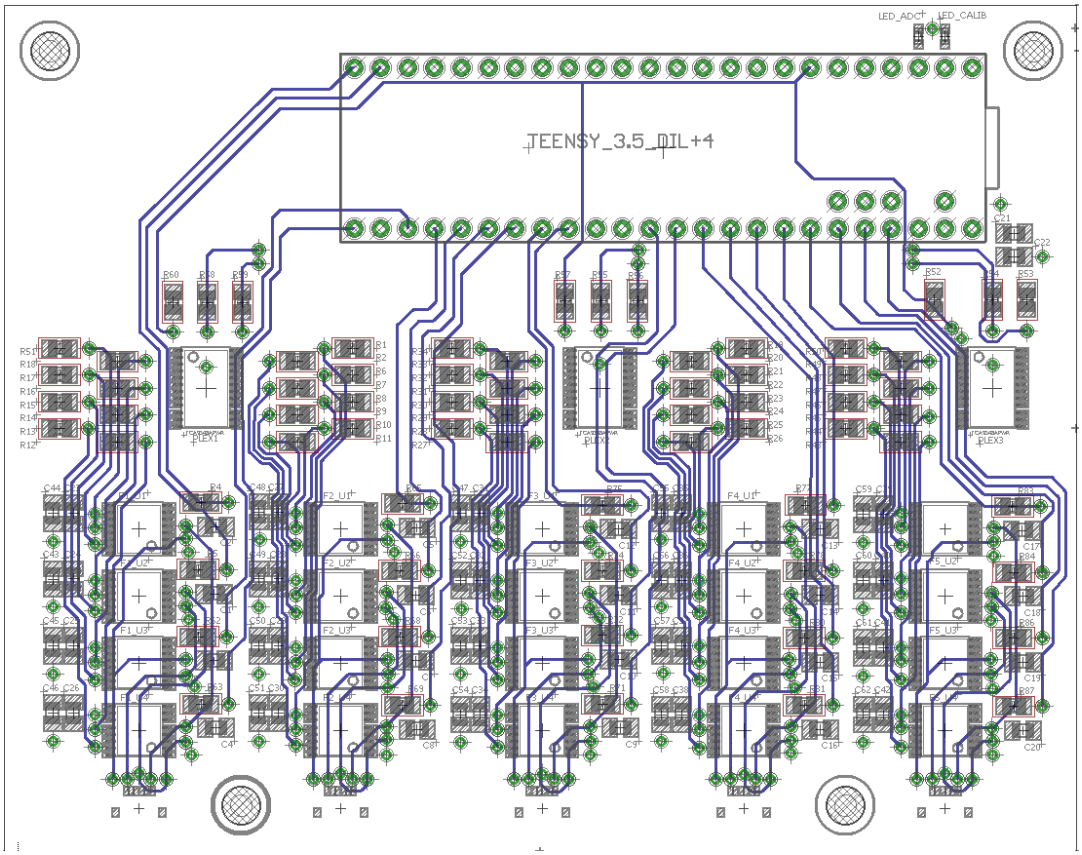


Figure B.5: The bottom trace layer view of the PCB is shown only.

Appendix C

HID command table

buffer[0]	buffer[1]	buffer[2]	buffer[3]	buffer[4]	buffer[5]	Task
d	g	N/A	N/A	N/A	N/A	Returns ADC mode
	t	N/A	N/A	N/A	N/A	Toggles ADC mode
m	g	N/A	N/A	N/A	N/A	Returns current verbosity mode
	s	verbosity	N/A	N/A	N/A	Sets how much diagnostic information, or verbosity, is sent to user's console
g	g	finger	N/A	N/A	N/A	Returns total gain associated with finger specified in buffer[2]
	s	user_gain[0]	user_gain[1]	finger	N/A	Sets all PGA units associated with finger specified in buffer[4] with value in user_gain
s	g	N/A	N/A	N/A	N/A	Returns the sampling rate of the device in kHz
	s	sampF[0]	sampF[1]	sampF[2]	sampF[3]	Sets the sampling rate by value in sampF
p	g	N/A	N/A	N/A	N/A	Returns the sampling period of the device in ms
	s	sampP[0]	sampP[1]	sampP[2]	sampP[3]	Sets the sampling period by value in sampP
c	N/A	N/A	N/A	N/A	N/A	Reruns the calibration routine
l	N/A	N/A	N/A	N/A	N/A	Toggles calibration LED
e	N/A	N/A	N/A	N/A	N/A	Sends an empty HID message
r	N/A	N/A	N/A	N/A	N/A	Echos the user message back
t	s	N/A	N/A	N/A	N/A	Sends an empty HID message
	u	N/A	N/A	N/A	N/A	Sends a uint8_t value (2)
	v	N/A	N/A	N/A	N/A	Sends an unsigned int value (514)
	w	N/A	N/A	N/A	N/A	Sends an unsigned long value (33686018)
	x	N/A	N/A	N/A	N/A	Sends a long value (-2)
	y	N/A	N/A	N/A	N/A	Sends a uint16_t value (513)
	z	N/A	N/A	N/A	N/A	Sends a float value (6.0)
	a	N/A	N/A	N/A	N/A	Sends a unsigned long value (1086324736)
	g	user_gain[0]	user_gain[1]	N/A	N/A	Sends back gain specified in user_gain to user

Table C.1: Table of HID commands This tabular list maps corresponding device behavior and output based on the user input character buffer. This list states all important commands, but is not exhaustive.

Appendix D

Full Results from PGA Setting Changes

V_{ref}	$V_{f,ratio}$	V_f	$V_{c,ratio}$	V_c	GI	GD	GO	$V_{in,1}$	$V_{in,2}$	V_{diff}	V_{probe}	ADC	V_{ADC}	V_{calc}	$\%err_{ADC}$	$\%err_{calc}$
3.27	0.25	0.82	0	0	4	0.5	2	1.05	0.76	0.29	2	40200	2.01	1.98	0.29	-1.125
3.27	0.5	1.64	0	0	4	0.5	2	1.05	0.76	0.29	2.82	56500	2.82	2.80	-0.03	-0.89
3.27	0.25	0.82	0	0	4	0.33	2	1.05	0.76	0.29	1.33	26700	1.33	1.30	0.17	-1.87
3.27	0.25	0.82	0	0	4	1	2	1.33	1.22	0.11	2.49	50100	2.50	2.52	0.39	1.00
3.27	0.25	0.82	-5	-0.014	4	0.5	2	0.75	0.46	0.29	2	40200	2.01	1.92	0.29	-3.90
3.27	0.25	0.82	0	0	8	0.5	2	2.15	1.89	0.26	2.75	55200	2.75	2.90	0.16	5.36
3.27	0.25	0.82	0	0	4	0.5	3	2.15	1.89	0.26	2.68	53800	2.68	2.79	0.16	3.96
3.27	0.25	0.82	0	0	16	0.5	2	1.96	1.89	0.07	1.89	38000	1.90	1.94	0.32	2.51
3.27	0.25	0.82	0	0	4	0.5	2.4	1.96	1.89	0.07	1.33	26200	1.31	1.32	-1.71	-0.98

Table D.1: Full results from PGA setting changes The full results of the amplifier setting change test are shown with ADC and transfer function errors compared to multimeter readings.

Appendix E

Full Results from Adjusting ADC Settings

Sampling Speed	Conversion Speed	Buffer Size (integers)	Resolution (bits)	Averaging Interval (samples)	Sampling period (ms)	Sampling Rate (kHz)
Medium	Medium	20000	16	8	756	26.46
Medium	Medium	40000	16	4	858	46.62
Medium	Medium	40000	16	8	>1000*	< 40.00*
High	High	20000	16	4	383	52.22
Very low	Very low	20000	16	4	>1000*	< 20.00*
Very low	Very low	20000	16	0	336	59.52
Slow	Slow	20000	16	0	289	69.20
Slow	Slow	20000	12	0	263	76.05
Medium	Medium	20000	12	0	150	133.33
Very high	Very high	20000	12	0	103	194.17
High	High	20000	12	0	103	194.17
Medium	Medium	20000	16	16	1405	14.23
High	High	20000	16	16	707	28.29
High	High	20000	32	32	937	21.34

Table E.1: Full results of ADC adjustment protocol All trials in this test were conducted to quantify noise and sampling rate under different speed and averaging settings. *Indicates reading outside range of measurement.

Appendix F

All ADC Reading Graphs Generated for Noise Measurement

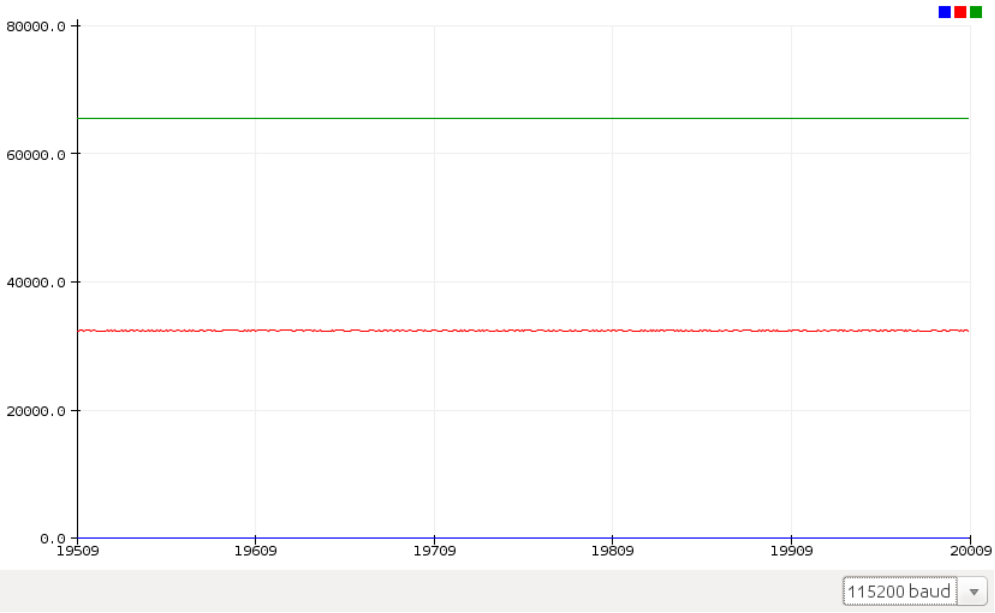


Figure F.1: Zero-force ADC reading The offset in the transfer function is set such that zero reading is roughly equivalent to 50% of reference voltage. (X-axis is samples since start of ADC collection; y-axis is digitized 16-bit voltage reading out of 65535; predicted upper axis limit is in green, predicted lower axis limit is in blue, and output reading is in red).

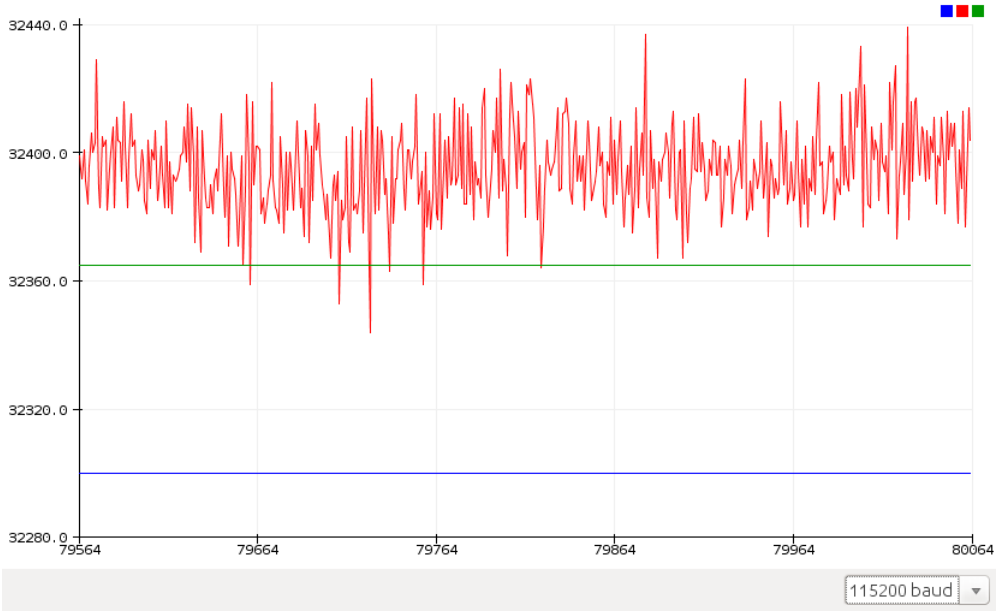


Figure F.2: Zoomed-in view 1 0-sample averaging, medium sampling speed, medium conversion speed

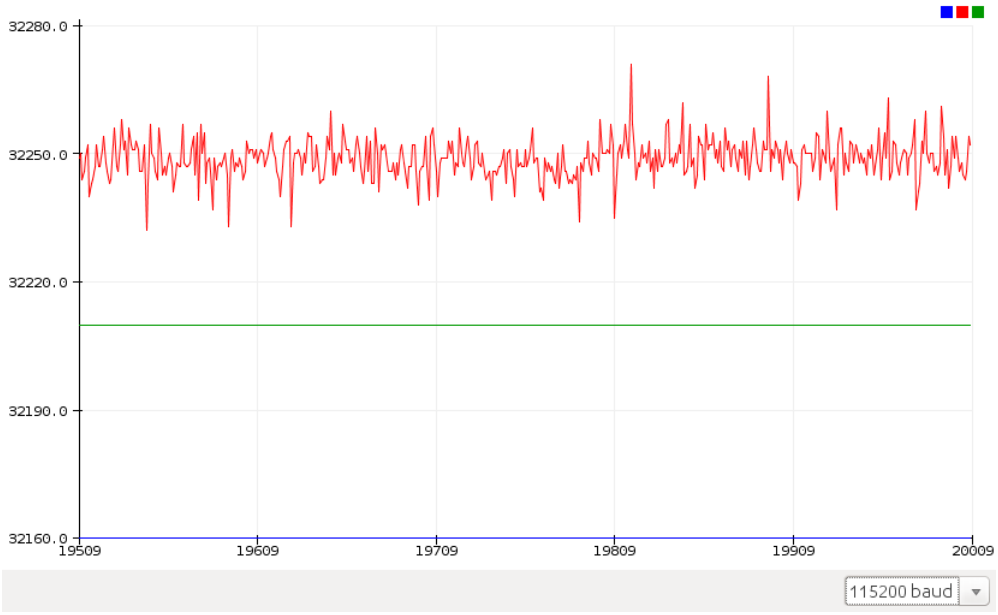


Figure F.3: Zoomed-in view 2 16-sample averaging; high sampling speed; high conversion speed

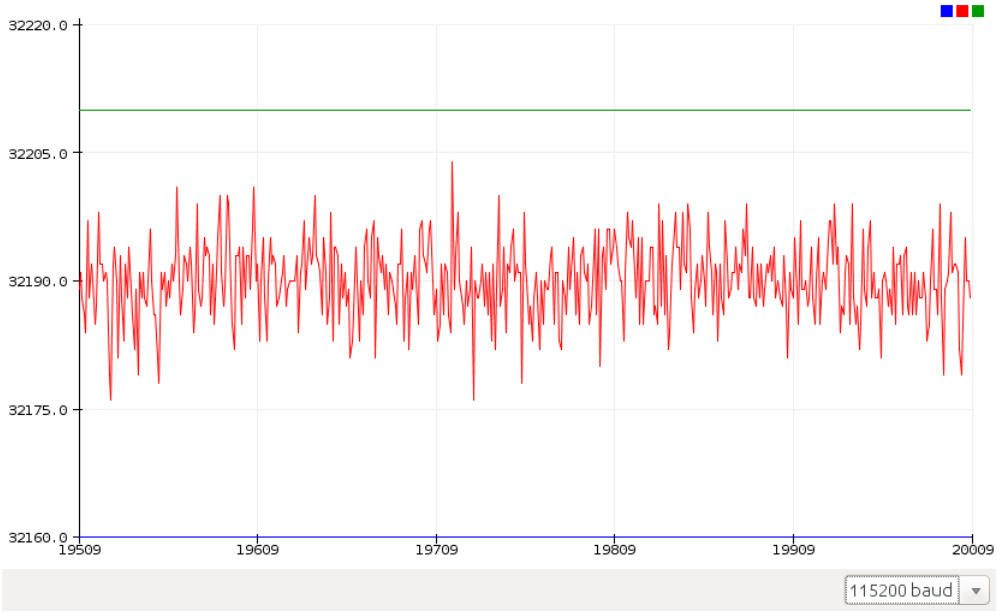


Figure F.4: Zoomed-in view 3 16-sample averaging; medium sampling speed; medium conversion

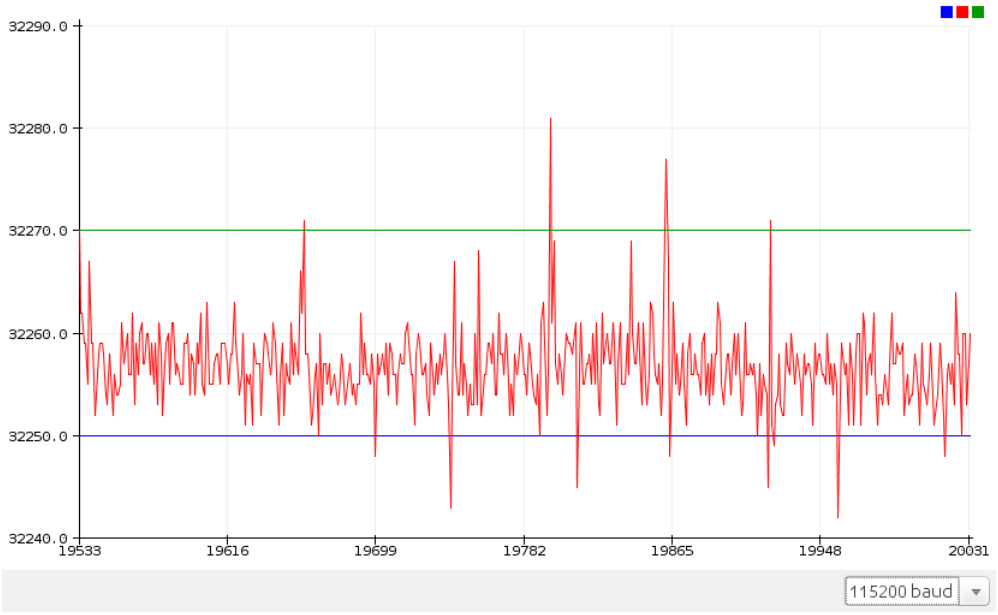


Figure F.5: Zoomed-in view 4 32-sample averaging; high sampling speed; high conversion speed

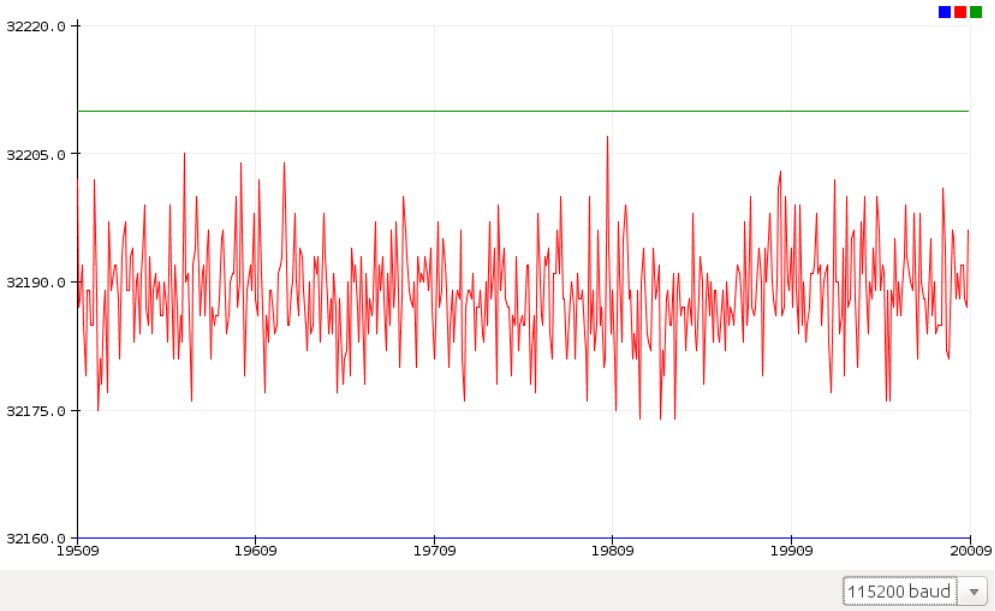


Figure F.6: Zoomed-in view 5 8-sample averaging; medium sampling speed; medium conversion speed

The following graphs were recorded much earlier, but provided little value for selecting optimal settings due to visual indication that noise seemed higher. Analysis also provided hard since graphs are zoomed out and incorporated an earlier implementation of the LP filter, so comparison would have little meaning.

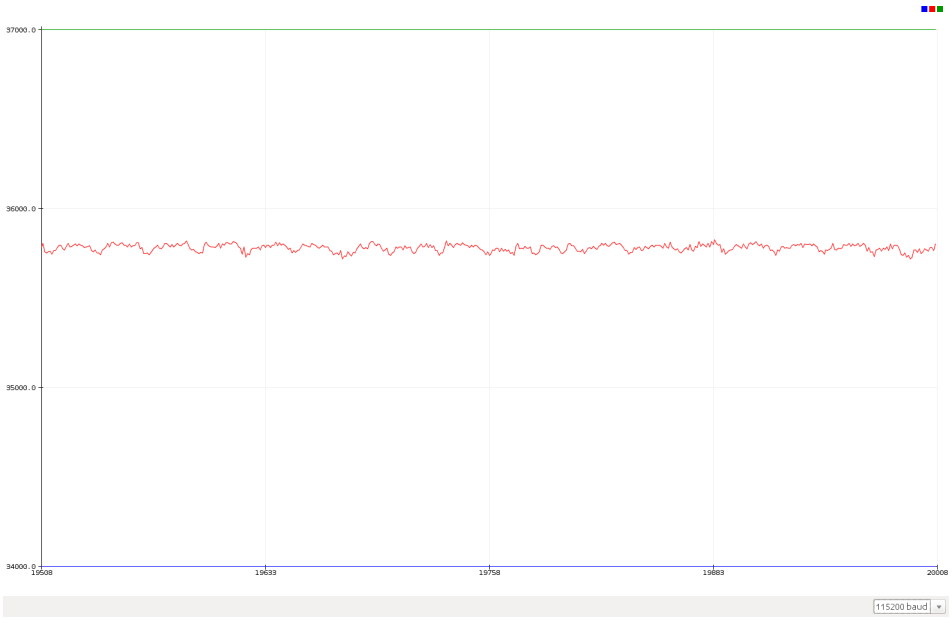


Figure F.7: Zoomed-in view 6 (Legacy) 4-sample averaging; slow sampling speed; slow conversion rate

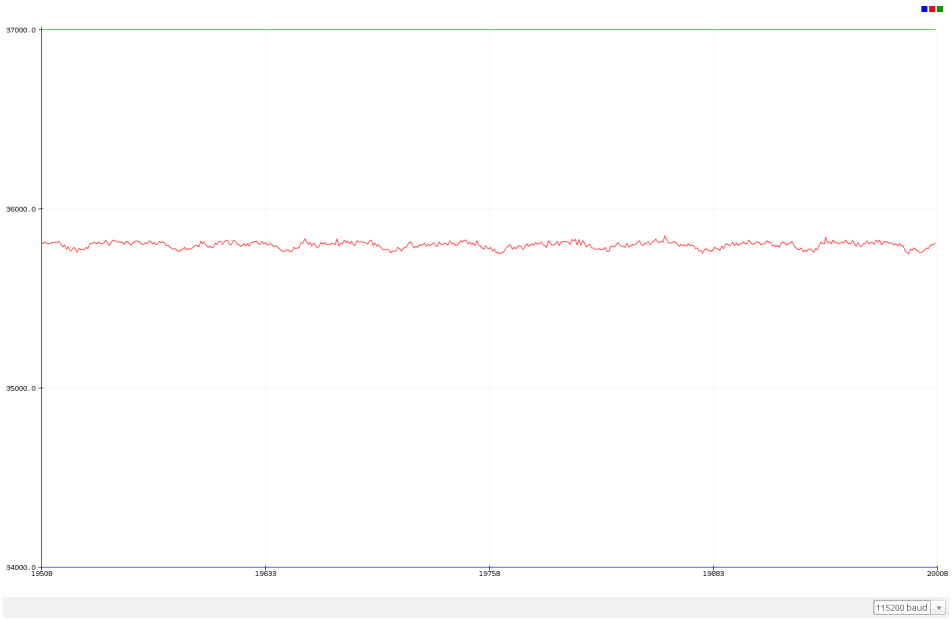


Figure F.8: Zoomed-in view 7 (Legacy) 4-sample averaging; medium sampling speed; medium conversion speed

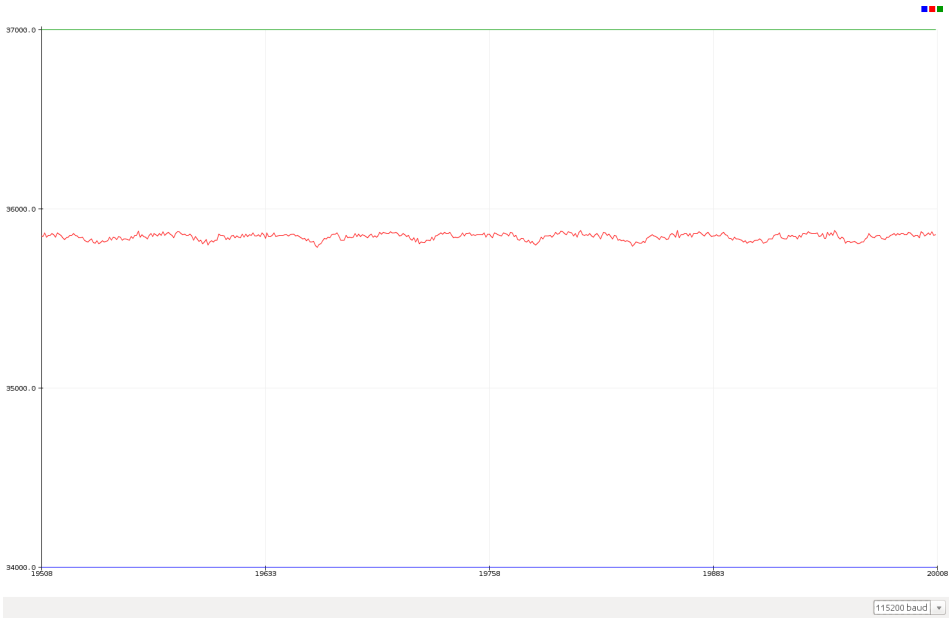


Figure F.9: Zoomed-in view 8 (Legacy) 8-sample averaging; high sampling speed; high conversion speed

Appendix G

Sample data collection output in Serial Monitor

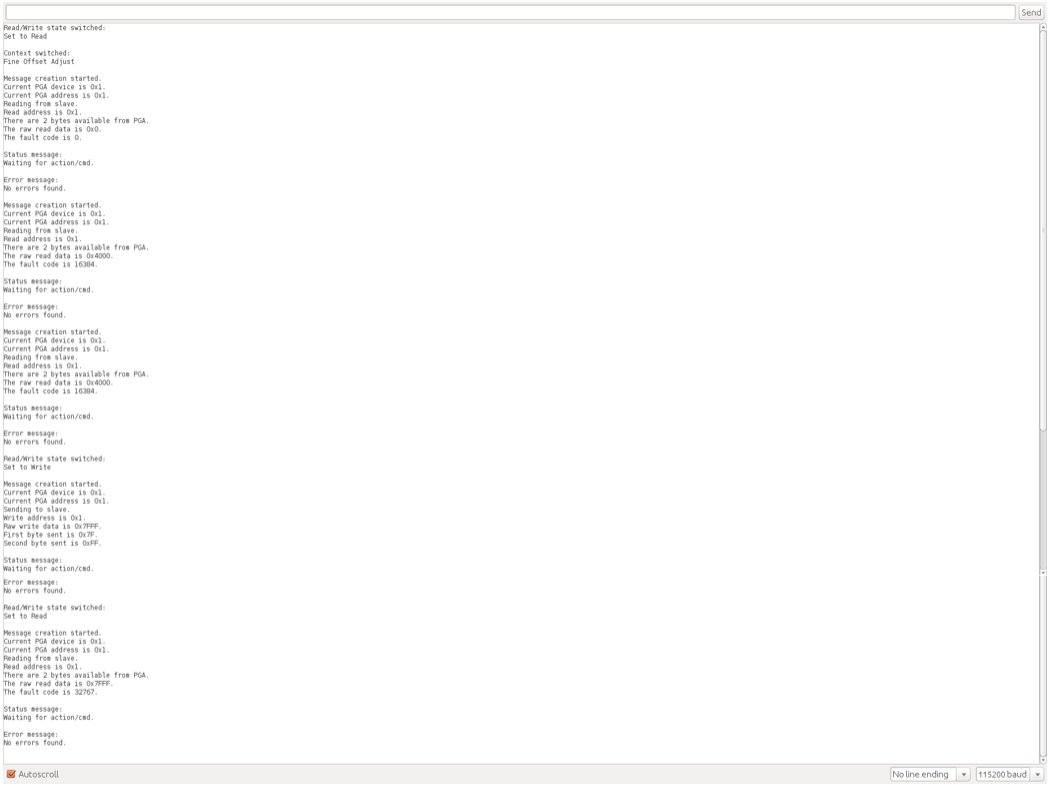


Figure G.1: pga_i2c library output An example console output showing that the Teensy is reading default values from a PGA unit and having one of its registers' value changed.

Appendix H

Interval regions sampled for force-voltage unit comparison

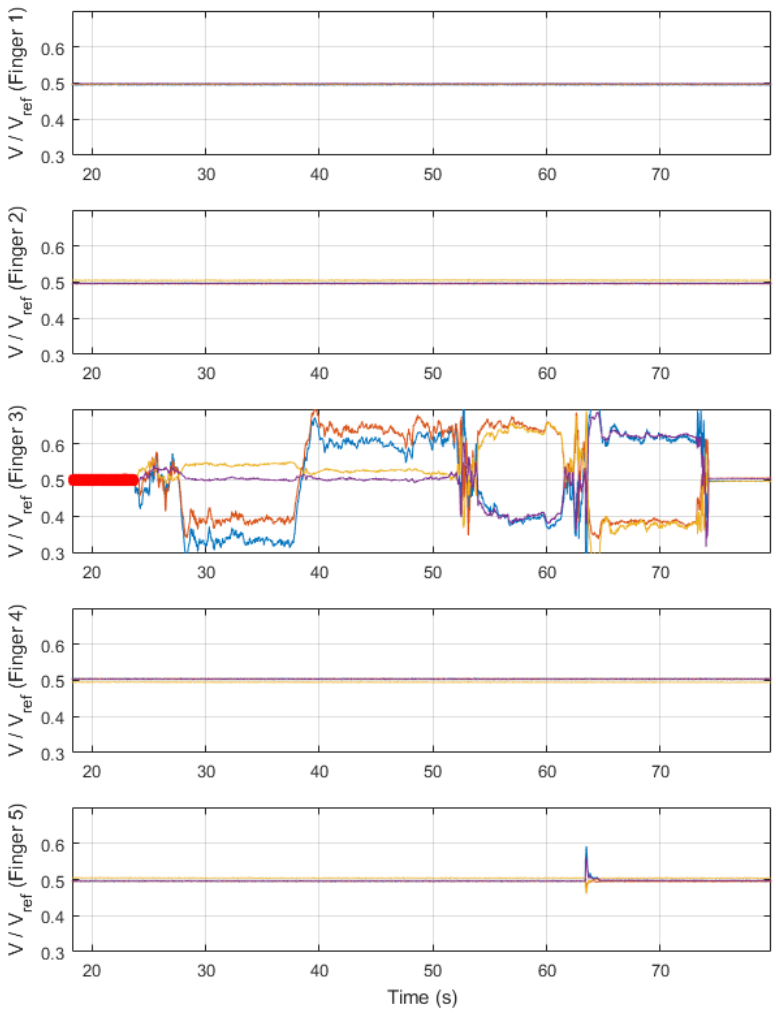


Figure H.1: Zero-force brush tool selection Voltage data points are selected in MATLAB that correspond to an absense of force application.

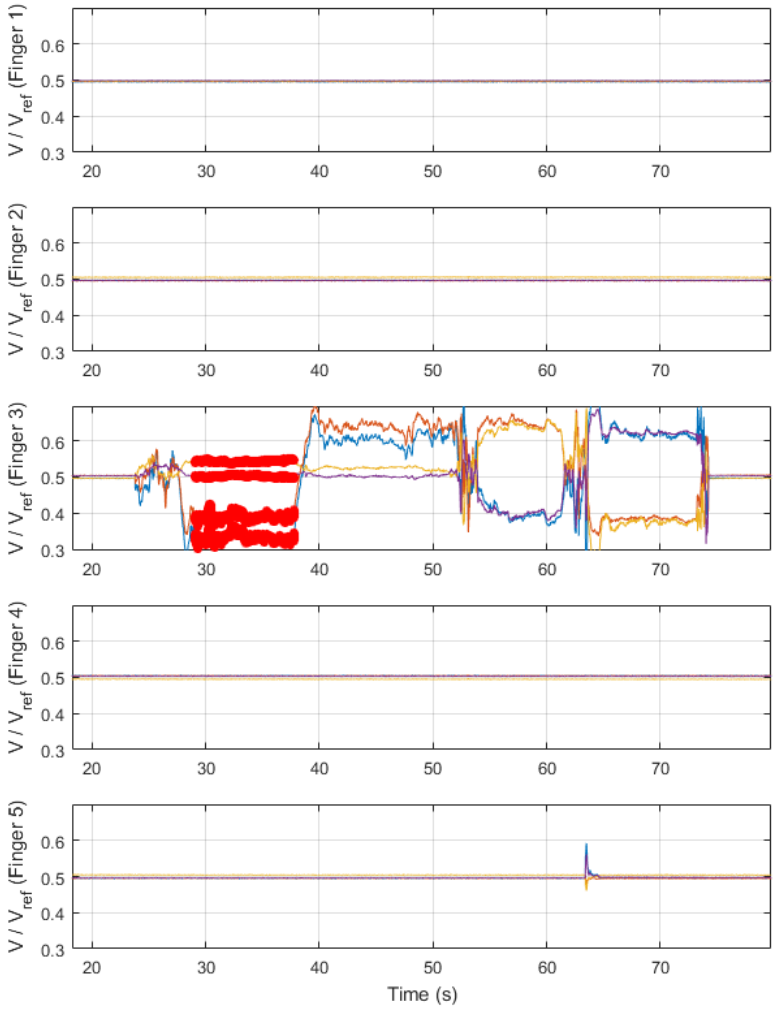


Figure H.2: Brush tool selection of 1 N in -Y direction Voltage data points are selected in MATLAB that correspond to a particular direction and magnitude of force. Cutoffs for the starting and end data points are determined by external timing of force and reasonable observation of the voltage traces.

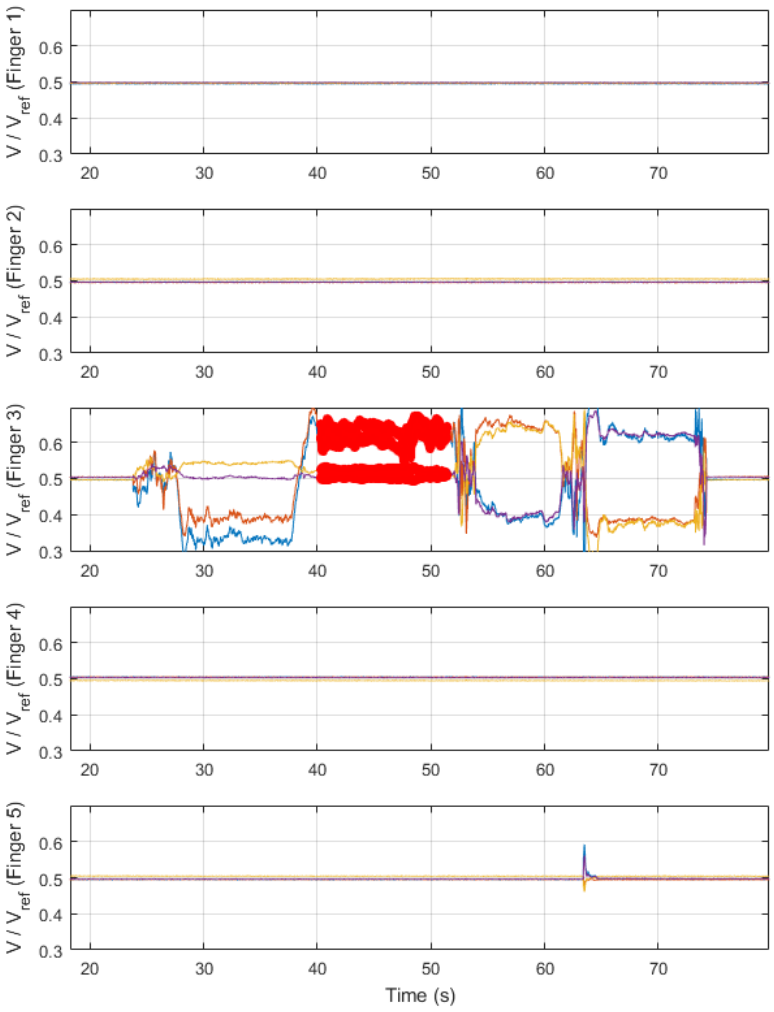


Figure H.3: Brush tool selection of 1 N in +Y direction

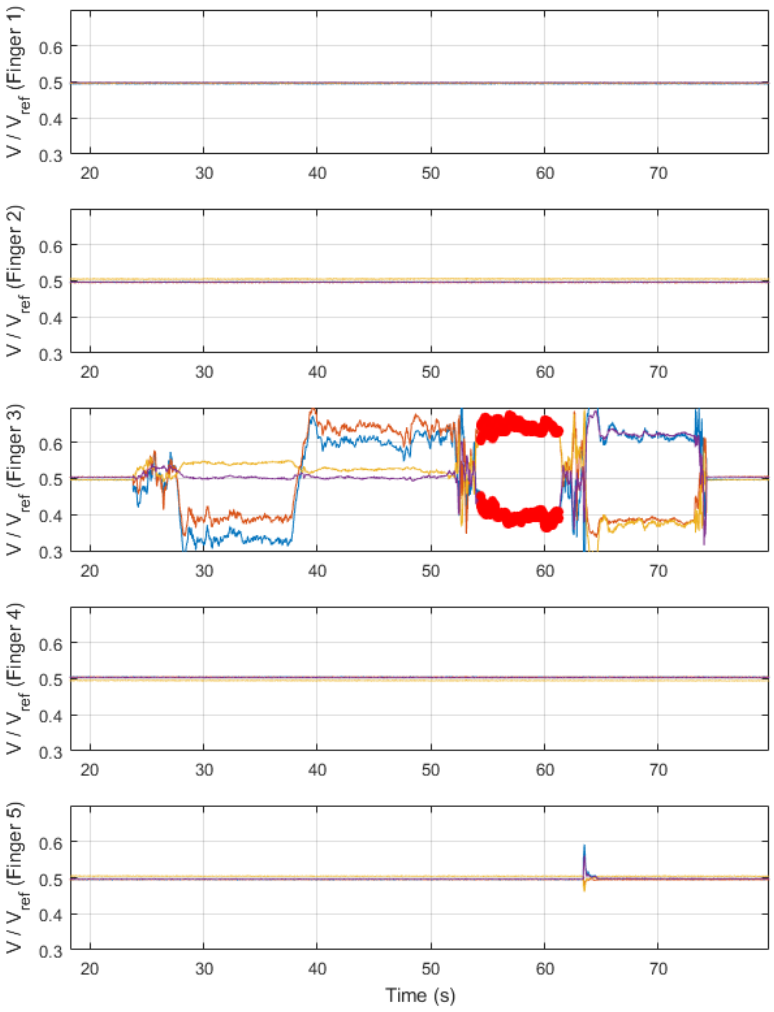


Figure H.4: Brush tool selection of 1 N in -X direction

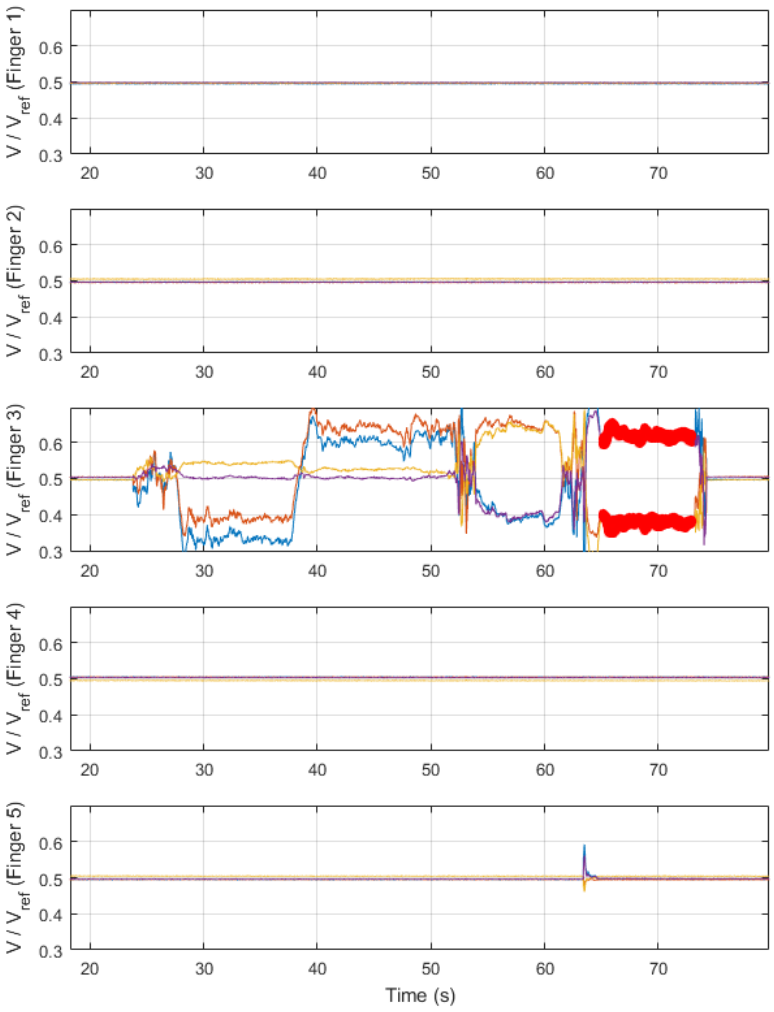


Figure H.5: Brush tool selection of 1 N in +X direction

Biography

Jacob Carducci was born in 1993 in the USA.

Jacob did undergraduate work at the Rose-Hulman Institute of Technology where he majored in Biomedical Engineering and Mechanical Engineering with a minor in Robotics. While there, he engaged in several academic projects including a self-monitoring beehive, a novel way of determining thermal properties from shoe outsole material, small ball levitation via adaptive fan control, and Android-based autonomous rover navigation. He also took active roles in the IGVC Robotics Team, Biomedical Engineering Society, Alpha Lambda Delta, and Tau Beta Pi honor societies.

In the summer of 2014, he worked in the Biomechanics Department of Life Fitness, where he assisted in the development of new exercise products through performance surveys of contact electrode sensors and gait analysis. In the summer of 2015, Jacob collaborated with the AMI Deployment Department of Commonwealth Edison to assist in the deployment of smart meters around the Chicagoland area.

In 2016, Jacob began pursuing a M.S.E. in Robotics at Johns Hopkins University. He was a lab assistant for Dr. David Kraemer's Robot Sensors and Actuators class. He now currently assumes lab positions in Dr. Jeremy

Brown's Haptics and Medical Robotics Lab and Dr. John Krakauer's Brain, Learning, Animation, and Movement Lab.